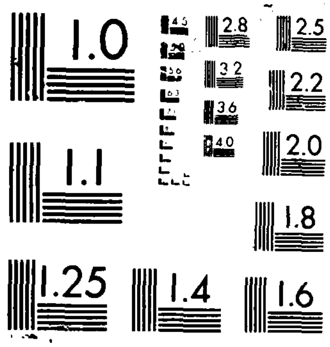
AD-A189 381 A GOVERNMENT-BINDING BASED PARSER FOR WARLPIRI A 1/2
FREE-WORD ORDER LANGUAGE..(U) MASSACHUSETTS INST OF
TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE L.. M B KASHKET
UNCLASSIFIED 22 SEP 87 AI-TR-993 N00014-85-K-0124 F/G 5/7 NL


1.0
2.8
2.5
3.2
2.2
3.6
2.0
4.0
1.1
1.8
1.25
1.4
1.6

(9)

Technical Report 993

A Government-Binding Based Parser for Warlpiri, a Free-Word Order Language

Michael B. Kashket

MIT Artificial Intelligence Laboratory

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER AI-TR-993	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) A Government-Binding Based Parser for Warlpiri, a Free-Word Order Language		5. TYPE OF REPORT & PERIOD COVERED Technical Report
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Michael B. Kashket		8. CONTRACT OR GRANT NUMBER(s) N00014-85-K-0124
9. PERFORMING ORGANIZATION NAME AND ADDRESS Artificial Intelligence Laboratory 545 Technology Square Cambridge, MA 02139		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Advanced Research Projects Agency 1400 Wilson Blvd. Arlington, VA 22209		12. REPORT DATE September 22, 1987
		13. NUMBER OF PAGES 167
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office) Office of Naval Research Information Systems Arlington, VA 22217		15. SECURITY CLASS. (of this report) UNCLASSIFIED
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Distribution is unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

None

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

principle-based parsing
Government-Binding theory
free word order

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Free-word order languages have long posed significant problems for standard parsing algorithms. This report presents an implemented parser, based on Government-Binding theory, for a particular free-word order language, Warlpiri, an aboriginal language of central Australia.

The words in a sentence of a free-word order language may swap about relatively

DD FORM 1473, 1 JAN 73 EDITION OF 1 NOV 65 IS OBSOLETE
S/N 0102-014-6601

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

20., continued

freely with little effect on meaning. It is assumed that this similarity in meaning is directly reflected in the syntax. The parser presented here properly processes free word order because it assigns the same syntactic structure to the permutations of a single sentence. The parser also handles fixed word order, as well as other phenomena.

This parser differs from more traditional rule-based parsing systems, e.g., context-free parsers, in that parsing is carried out via the construction of two different structures, one encoding precedence information and one encoding hierarchical information. This bipartite representation is the key to handling both free- and fixed-order phenomena.

A GOVERNMENT-BINDING BASED PARSER FOR WARLPIRI, A FREE-WORD ORDER LANGUAGE

by

Michael B. Kashket

This is a revised version of a thesis submitted to the Department of Electrical Engineering and Computer Science on January 30, 1987 in partial fulfillment of the requirements for the degree of Master of Science.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense (DARPA) under Office of Naval Research (ONR) contract N00014-85-K-0124; in part by a General Electric fellowship; in part by DARPA under ONR contract N00014-80-C-0505; and in part by a National Science Foundation grant, 8552543-DCR, under a Presidential Young Investigator Award to Professor Robert C. Berwick.

Abstract

Free-word order languages have long posed significant problems for standard parsing algorithms. This thesis presents an implemented parser, based on Government-Binding (GB) theory, for a particular free-word order language, Warlpiri, an aboriginal language of central Australia.

The words in a sentence of a free-word order language may swap about relatively freely with little effect on meaning; the permutations of a sentence mean essentially the same thing. It is assumed that this similarity in meaning is directly reflected in the syntax. The parser presented here properly processes free word order because it assigns the same syntactic structure to the permutations of a single sentence. The parser also handles fixed word order, as well as other phenomena. On the view presented here, there is no such thing as a "configurational" or "non-configurational" language. Rather, there is a spectrum of languages that are more or less ordered.

The operation of this parsing system is quite different in character from that of more traditional rule-based parsing systems, *e.g.*, context-free parsers. In this system, parsing is carried out via the construction of two different structures, one encoding precedence information and one encoding hierarchical information. This bipartite representation is the key to handling both free- and fixed-order phenomena.

This thesis first presents an overview of the portion of Warlpiri that can be parsed. Following this is a description of the linguistic theory on which the parser is based. The chapter after that describes the representations and algorithms of the parser. In conclusion, the parser is compared to related work. The appendix contains a substantial list of test cases—both grammatical and ungrammatical—that the parser has actually processed.

Thesis Supervisor: Robert C. Berwick

Title: Associate Professor of Computer Science and Engineering

Acknowledgments

I wish to thank my thesis advisor, Robert Berwick, for his helpful advice and criticisms. I also wish to thank Mary Laughren for her instruction on Warlpiri, without which I would not have been able to create this parser.

This thesis is dedicated to my parents, Eva and Shelby Kashket.

Contents

Chapter 1

Introduction

This thesis presents a solution for the previously unsolved problem of parsing free-word order languages.[1,2] In these languages, the words in a sentence may swap about relatively freely with little effect on meaning; the permutations of a sentence mean essentially the same thing. It is assumed that the similarity in meaning is directly reflected in the syntax. So, a parser that properly processes free word order must assign the same syntactic structure to the permutations of a sentence. The parser also handles fixed word order, as well as other syntactic phenomena.

Until recently, many natural-language parsers have been designed around computationally attractive formalisms, such as context-free grammars, that have little linguistic foundation. To date, these parsers have worked correctly on but a limited subset of natural utterances. However, they have arrived at their results quite quickly. The theory of parsing presented in this thesis, on the other hand, is based on one current linguistic theory. The result is that the implemented parser outputs linguistically meaningful structures corresponding to the input sentence. The main hypothesis here is that we are more likely to arrive at a successful parser if we base it on linguistic theory, rather than on computational considerations alone.

The parsing model must, of course, be tested on a natural language. Warlpiri, an aboriginal language from central Australia, was chosen because it is perhaps the paradigmatic natural language exhibiting free word order. On a more practical level, Warlpiri has a relatively simple syntax, and a fairly small lexicon, which makes for an easier job of producing a parser that handles an appreciable subset of the language. Finally, there has been a good deal of linguistic inquiry into the language (see, for example, [Lau78], [Hal83], [Sim83], [Nas86]), which increases the chances that a parser based on this theory will actually perform well.

[1] Johnson[Joh85] has written a parser based on Definite Clause Grammar that covers the extreme string permutation found in free-word order languages. The parser is written in a general proof system, and thus suffers from a lack of explanatory power, which Johnson does acknowledge. A parser that provides a true solution to this problem must output linguistically motivated structures.

[2] Lexical-Functional Grammar seems to provide a well-motivated analysis of free-word order phenomena; in fact, it is very similar to the theoretical basis of the parser presented here. However, I am unaware of any parser based on LFG that processes free-word order languages. See the concluding chapter for more remarks on LFG.

Here is a concrete example from the target language of the parser: consider (1).[3,4]

(1) *Ngajulu-rlu ka-rna-rla punta-rni kurdu-ku karli.*
I-ERG IMPERF-1s-3d take-NONPAST child-DAT boomerang
'I am taking the boomerang from the child.'

The first word of (1), *ngajulu-rlu*, is the subject; the last word, *karli*, is the object; and the fourth word, *kurdu-ku*, is the indirect object. The *grammatical functions* (*e.g.*, subject, object, and indirect object) of these words are determined by their case-markings (*e.g.*, *-rlu* of *ngajulu-rlu*), and not by their positions, as in, say, English. This is exemplified by the sentences in (2) which are equivalent ways of saying (1).[5] In these sentences the nouns move about freely. Notice that the verb, *punta-rni*, appears in different positions as well; although not demonstrated here, it may also begin the sentence.

(2) a. *Karli ka-rna-rla punta-rni ngajulu-rlu kurdu-ku.*
boomerang IMPERF-1-3 take-NONPAST I-ERG child-DAT
'It is the boomerang I am taking from the child.'

b. *Kurdu-ku ka-rna-rla ngajulu-rlu karli punta-rni.*
child-DAT IMPERF-1-3 I-ERG boomerang take-NONPAST
'From the child I am taking the boomerang.'

c. *Ngajulu-rlu ka-rna-rla punta-rni karli kurdu-ku.*
I-ERG IMPERF-1-3 take-NONPAST boomerang child-DAT
'I am taking the boomerang from the child.'

There is, however, an ordering constraint shown in these examples. The auxiliary word, *ka-rna-rla*, must appear in the second position.[6] Even given the fixed position of the auxiliary, these four permutations do not exhaust the possibilities for uttering this sentence. There are in fact 4!, or 24, different ways of saying the same thing.

So far we have been talking about meaning, and not the parser's output domain, syntax. The claim here is that the aspects of meaning that remain constant across word permutation are directly mirrored in syntax. That is, the ordering of the words is independent of their grammatical function, which is later interpreted as part of the meaning of the sentence. In order for a parser to properly handle free word order, it must output the same syntactic structure for each of the permutations of

---

[3]Until recently, there was no written Warlpiri. Now the Warlpiri are being taught a written system that uses Roman characters. There is also a standard orthography for the phonemes that do not appear in English, *e.g.*, *ng* is used to denote a palatalized nasal, similar to the English *sing* (See [Nas86] for a complete description.) The hyphens in the examples are not part of the standard written system; they are included only to aid the novice reader.

[4]In this sentence, the objects, *kurdu-ku* and *karli*, are given a definite reading. In general, this information is unavailable from the sentence itself and must be gleaned from context. For simplicity, definite reference will be used.

[5]There is some difference among the sentences, of course, but it concerns a change in focus, rather than a change in meaning. The first word is given a slight emphasis over the others. This subtle difference is mimicked in the English translations for the sentences.

[6]Actually, it may appear in the first position too. The details are rather complex, and they are described below.

a single sentence. The result of parsing (1) and its permuted cousins should yield structures that encode the grammatical functions shown in (3).

(3)	subject	*ngajulu* ('I')
	object	*karli* ('boomerang')
	indirect object	*kurdu* ('child')

The ability to parse such examples has eluded previous parsing systems. Their difficulty with free word order can be demonstrated with a very common parsing technology, context-free parsing, that arose from compiler design.[7] Context-free parsers are based on context-free grammars, consisting of a set of rewrite rules. These rules contain a left-hand side and a right-hand side. On the left is a single non-terminal symbol which may be replaced with the string of symbols (terminal and non-terminal) on the right. Given a special start symbol, the grammar is said to derive a string if there is some sequence of rewrites that results in a sequence of terminal symbols that matches the string. The language of such a grammar is the set of strings that it can derive through all possible sequences of rewrites.

Context-free parsers suffer from two problems when it comes to parsing free-word order languages; both result from the nature of their underlying grammar formalism. The first problem is that extremely unperspicuous grammars must be written in order to cover word permutation. These grammars hide the regularity behind grammatical functions. Consider the grammar in (4) that covers a language containing exclusively transitive verbs (*i.e.*, sentences consist of a verb, a subject, and an object). With this grammar six rules are required to obscurely encode the fact that verbs take two nouns, one of which is the subject and one of which is the object.

(4)
$$S \rightarrow NP_s\ NP_o\ V$$
$$S \rightarrow NP_s\ V\ NP_o$$
$$S \rightarrow V\ NP_s\ NP_o$$
$$S \rightarrow NP_o\ NP_s\ V$$
$$S \rightarrow NP_o\ V\ NP_s$$
$$S \rightarrow V\ NP_o\ NP_s$$

These parsers have a more significant failing. The structures that they output are not linguistically precise because they do not make explicit important syntactic relations. The sample grammar does not, for instance, highlight the grammatical functions of subject and object. A better grammar would encode this information directly, such as the hierarchical grammar given in (5). Here the subject is the sibling of the verb phrase (denoted 'VP') and the object is the sibling of the verb.

(5)
$$S \rightarrow NP_s\ VP$$
$$VP \rightarrow V\ NP_o$$

[7] There are several other natural language parsers in the literature which, although they are not necessarily based on context-free rules, still lack the ability to handle free word order. I discuss them briefly in the concluding chapter.

However, this grammar suffers from inadequate coverage. Even removing the ordering of the elements of the right-hand side, the grammar does not generate either of the sentence schemata found in (6).

(6) $V\ NP_s\ NP_o$
$NP_o\ NP_s\ V$

My solution for the problem of parsing free word order is based on Government-Binding (GB) theory.[8] GB is a linguistic theory that is concerned with the syntax of a single sentence. The structures that it provides will, if it is correct, make the important linguistic information explicit. However, GB is not a computational theory. It does not specify *how* parsing (or generation, for that matter) is to be done: it only specifies *what* the underlying syntax is to be. By basing the parser on GB, I mean that its output is dictated by GB theory, and furthermore, that the operations of the parser follow the modularity of the linguistic theory; this will be elaborated below.

In fact, the parser computes only a part of GB output representations. GB consists of several levels of representation, each of which encodes information relevant to a certain aspect of the sentence. The parser produces two output structures based on these levels. Precedence structure (PS) represents a part of so-called Phonological Form (PF), as well as part of the theory of morphology (word structures), while syntactic structure (SS) represents so-called S-structure. The theory behind these structures is quite complex; they will be described in greater detail in the next chapter. For our purposes here, however, we can give an abbreviated description that will serve to show how the linguistic theory provides the proper structures for handling free word order.

Precedence structure is used to represent the precedence information inherent in the input, which I take to be a slightly processed version of the speech stream. It is assumed that the speech stream has been broken down into its constituent morphemes, words, and phrases, upon which there is a total ordering by virtue of the linear nature of speech. PS represents the ordering of the input that is relevant to syntax; thus it is a partial ordering. For example, in the PS for *ngajulu-rlu* there would be an ordered pair of morphemes (*ngajulu*, *rlu*), as the ordering of the noun followed by the case-marker is syntactically relevant. In fact, it is ungrammatical to reverse the order of the morphemes; there is no such word as **rlu-ngajulu* in Warlpiri. On the other hand, the PS for Warlpiri would not contain relations between the words, because their order does not matter.[9]

The other part of the parser's output is syntactic structure. Unlike PS, SS has no precedence information encoded into it: it is a strictly hierarchical structure. This differs from traditional GB theory, where S-structure is an ordered level of representation. The argument for removing this information relies on Occam's razor: there doesn't seem to be a need for precedence information at the level of S-structure

[8] See, for example, [Cho81], [Cho82].

[9] This is also a bit of a simplification. Word ordering within phrases does matter; this is handled by the parser, and will be described in the following chapters. It is really the phrases that may permute. Perhaps it would be more nearly accurate to call Warlpiri a free-*phrase* order language.

because any ordering that is required may, and must, be represented at PF. For now, we can take SS simply as hierarchical. SS is where such relations as grammatical functions would be represented. Subjects are taken to be the siblings of VPs, and objects are taken to be siblings of V.

Let's see how the GB grammar would account for the simple language of transitive verbs in Warlpiri. Grammatical function is identified by the *case-markings* on nouns. For one class of verbs, the subject is marked with the ergative case, often appearing as the suffix *-rlu*; the object is marked with the absolutive case, which is not overt phonologically.[10] As has already been mentioned, case-markers must be enclitic (*i.e.*, affixed) to the nouns that they mark, and they must be to their right. The nouns so marked receive the case of their case-marker. Verbs must be inflected for tense by tense markers that are enclitic to the stem and to their right. These facts are encoded in the PS for Warlpiri, as shown in (7).

(7) PS: N is followed by C (intraword)
V is followed by T (intraword)

The grammatical functions of the nouns are represented in SS. As with the improved grammar above, we consider a sentence to consist of a noun phrase, the subject, and a verb phrase. The VP, in turn, consists of the verb and a noun phrase, its object. The SS for the simple Warlpiri sentences is given in (8).

(8) SS: S dominates NP_s and VP
VP dominates V and NP_o

Now consider (9) which is an abbreviated version of the sample sentence, (1) above.[11]

(9) *Ngajulu-rlu punta-rni karli.*
I-ERG take-NONPAST boomerang
'I am taking the boomerang.'

The PS returned by the parser encodes the three orderings relevant to syntax, all of which are intraword: see figure 1.1. In this graphic depiction of the PS, nodes representing categories are connected where ordering is relevant. Hence the tree connecting the verb stem, *punta*, and the tense element, *rni*, for instance. Note that the ordering between the words is not represented in the PS for the sentence because it is not important.

The SS that is returned contains the hierarchical structure for the verb and both its subject and object. See figure 1.2. Remember that the graph here does not use precedence so it could have been depicted with the subject node and the verb phrase node in the other order, likewise for the verb node and the object node.

[10] The ergative and absolutive cases are the analogs of the nominative and accusative cases found in languages like English. For the purposes of this thesis, there is no significant difference other than in the labeling of the case-markers.

[11] In fact, this sentence is ungrammatical because the null auxiliary word does not agree with subject of the sentence. More details of the auxiliary will be given below.

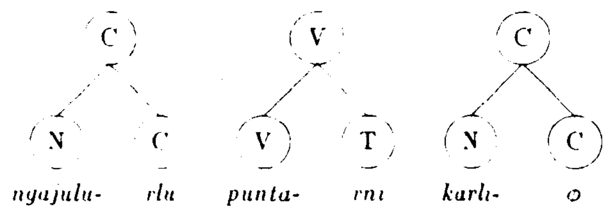


Figure 1.1: The PS for (9).

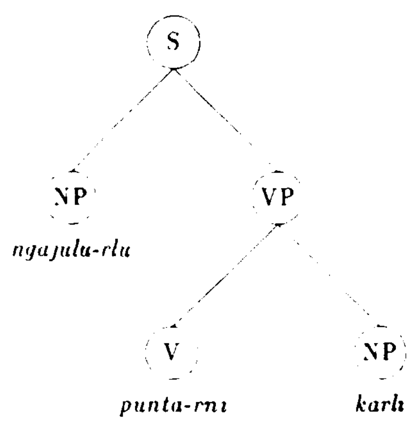


Figure 1.2: The SS for (9).

This bipartite representation also handles fixed word order, as in English.[12] In English, there are no overt case-markers for the subject or the object. Instead, the subject precedes the verb and the object follows. In this sense the verb itself acts as the case-marker. The noun phrase to its left is marked for nominative case, and the noun phrase to its right is marked for accusative case. As with Warlpiri, the verb is inflected for tense with a tense element that is enclitic to the verb stem and to its right. These facts are given in (10) which is the PS for this simple subset of English.

(10) PS: N_{nom} is followed by V (interword)
N_{acc} is preceded by V (interword)
V is followed by T (intraword)

As with Warlpiri, the grammatical functions of the noun phrases are represented in SS. In fact, the SS for English is the same as that for Warlpiri. The only difference concerns the mapping from case to grammatical function. In English, the noun phrase marked for nominative case is mapped to subject, and the noun phrase marked for accusative case is mapped to object. The SS for the simple English is given in (11).

(11) SS: S dominates NP_s and VP
VP dominates V and NP_o

The rough translation of (9) is given in (12). Let's examine the two structures returned by the parser for this sentence. The first, PS, encodes the ordering relations that are syntactically relevant; see figure 1.3. As stated in the PS for this subset of English, the verb stem must be followed by a tense element, in this case *-ing*. For now we will ignore the modal verb *am* as it is part of the auxiliary, which is not being covered by the simple grammar. The other part of the PS concerns the subject and object noun phrases, *I* and *the boomerang*, respectively. Note that in PS the verb, *taking*, is connected to the noun phrases as their relative ordering is relevant.

(12) I am taking the boomerang.

The SS that is returned contains the hierarchical structure for the verb and both its subject and object. See figure 1.4. It is equivalent to the SS for the Warlpiri sentence, as one would expect. Both sentences are saying the same thing, at least as far as the rough translation goes. Inasmuch as the meanings are equivalent we would expect to see identical syntactic structures, which the parser does indeed provide.

The difference between the so-called free-word order language, Warlpiri, and the fixed-word order language, English, then, is whether or not the predicator (for these simple sentences it is the verb) is also a case-marker. In both languages — and, it is believed, in all languages — case-marking is a directed relation. In Warlpiri, this

[12] The parser has not yet been tailored for English. The structures presented here are extrapolated from the theory underlying the parser and its performance on comparable Warlpiri phenomena.

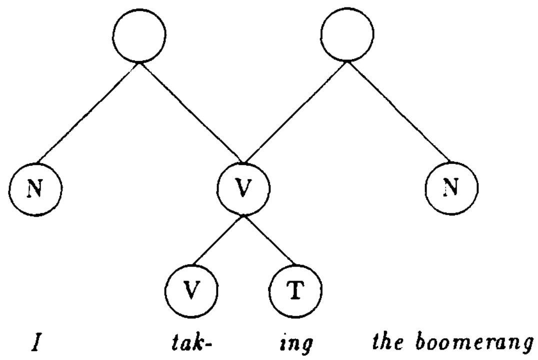


Figure 1.3: The PS for (12).

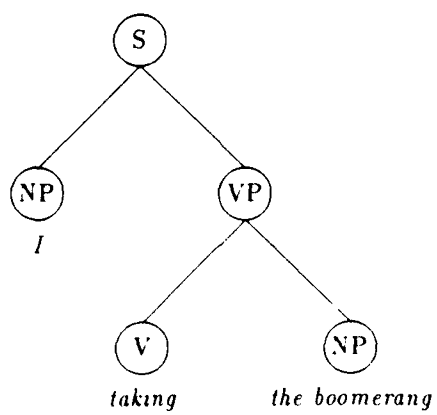


Figure 1.4: The SS for (12).

type of case-marking is performed by clitics, and in English by words. When the predicator is the case-marker, there is an ordering relation between the predicator and its arguments; when it is not, there is no such ordering relation. This distinction is directly reflected in the two output structures. It is only a question of which elements in PS perform the ordered action of case-marking. The syntactic relation of grammatical function—represented in SS—is unordered in both cases.

The terms "free order" and "fixed order" are a bit misleading, however. As noted above, Warlpiri does exhibit some ordering phenomena, for instance, among the morphemes of a word, and among the words of a phonological phrase. At the same time, English exhibits some free order. One common example is the ordering of prepositional phrases. Consider the sentences in (13) and (14) which mean essentially the same thing.

(13) I went to the store with Mary.

(14) I went with Mary to the store.

Prepositional phrases can be processed quite neatly by the parser. The ordered relation between the preposition and its object noun phrase is given in PS, shown in (15). The preposition is a case-marker, marking its object for its own case.

(15) PS: P is followed by NP (interword)

Syntactically speaking, prepositional phrases function as objects of the main verb, so, like object noun phrases, they are dominated by the VP node in SS:

(16) SS: VP dominates PP

The PSs for the sample sentences are given in figures 1.5 and 1.6. Note that the relevant ordering of the prepositions to the object noun phrases is, indeed, represented here, while the ordering of the verb and the prepositional phrases is not. The SS for both of the sentences is shown in figure 1.7.

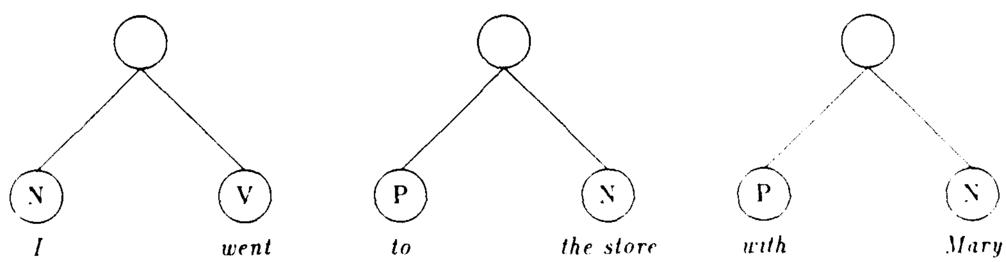


Figure 1.5: The PS for (13).

This mixture of fixed and free order seems to hold across languages. No language exhibits entirely free or fixed order; rather, languages lie along a spectrum where

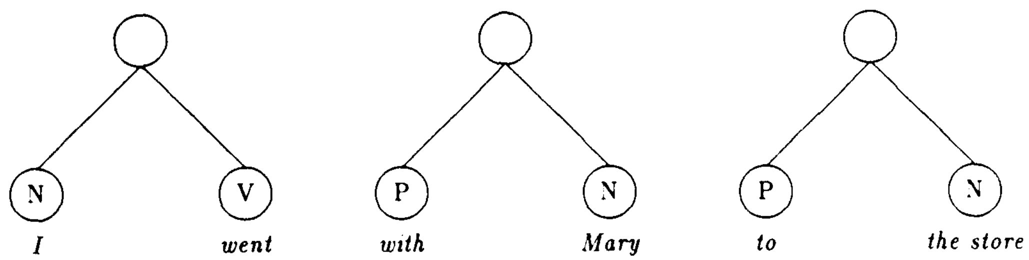


Figure 1.6: The PS for (14).

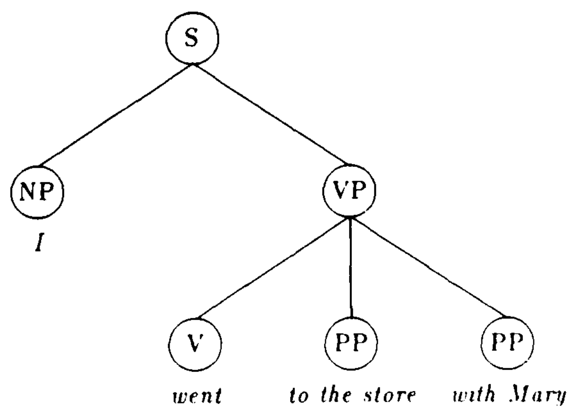


Figure 1.7: The SS for (13) and (14).

Warlpiri sits at one end and English at the other. This lends support for a bipartite representation that permits processing of both. The difference in processing particular languages will be reflected in which structure does which part of the parsing job. In more ordered languages, more of the ordering in PS will be relevant to syntax; in less ordered languages, less of the ordering will be important. In both cases, the syntactic structure will bear the responsibility for representing the grammatical function of the elements, as well as other syntactic relations.

Processing free- and fixed-order phenomena are not the only accomplishments of the parser. The contributions of the thesis are listed briefly in table 1.1. The parser joins a new and growing class of natural-language parsers based on Government-Binding theory.[13] A major task of this thesis was to precisely formulate GB theory in order to be able to compute its representations. In this process, I found it necessary to modify the theory of S-structure (explained below) to account for free word order. Given this account, the next task was to formulate a set of representations and algorithms that would compute the mapping from input sentence to output structures. Lastly, I implemented this design, and tested it on a range of inputs, both grammatical and ungrammatical, to ensure that it handled the phenomena properly.

- based the parser on Government-Binding theory
- modified the theory of S-structure
- designed the representations and algorithms
- implemented and tested the parser

Table 1.1: The contributions of the thesis.

Before introducing the parser formally, in the following section I outline the Warlpiri phenomena that the parser can handle.

1.1 A Warlpiri Primer

While there are over 2500 native speakers of Warlpiri, few of them will read this thesis. Therefore, it is useful to introduce some basic Warlpiri. If you are fluent in Warlpiri, feel free to skip to the next section.

1.1.1 The Major Parts of Speech

The primer begins with a discussion of the three major parts of speech: nouns, verbs, and auxiliaries. The analysis of nouns and verbs is relatively straightforward, but the auxiliary is more complicated. I begin with nouns.

Nouns must be declined, with the case-marker suffixing onto the noun. There are three syntactic cases in Warlpiri: ergative, absolutive, and dative. (17) gives

[13] There are a number of parsers based on other linguistic theories. There are also a handful of GB-based parsers. I discuss these other efforts in the concluding chapter.

some examples of declined nouns. (a) shows a noun marked for dative case. (b) shows a pronoun, which is declined just as nouns are. (c) shows a noun marked for absolutive case; there is no overt marker of this case.

(17) a. *kurdu-ku*
child-DAT
'child' marked for dative case

b. *ngajulu-rlu*
I-ERG
'I' marked for ergative case

c. *karli*
boomerang
'boomerang' marked for absolutive case

Nouns may be declined for number, in addition to case. There are four number-markers in Warlpiri: singular, dual, paucal,[14] and plural. Both dual and paucal have overt phonological elements which appear just after the noun, enclitic to it. (18) is an example of a noun that is overtly marked for number.

(18) *kurdu-jarra-ku*
child-DUAL-DAT
'to/from the two children'

Verbs must be inflected for tense and mood. This is accomplished by suffixing a tense element (which contains both tense and mood information) onto the verb. There are five tenses in Warlpiri: non-past, past, irrealis, future, and presentational. There are also five conjugation classes in Warlpiri, so there are a total of 25 inflections for verbs.[15] (19) presents a couple of examples of inflected verbs.

(19) a. *punta-rni*
take-NONPAST
'is taking'

b. *nya-ngu*
see-PAST
'saw'

The auxiliary word is analogous to the auxiliary verb system in English. Warlpiri's auxiliary consists of several component morphemes, each of which is optionally uttered. The first component is a complementizer which is null (*i.e.*, not uttered) for the simple declarative sentences that are the domain of the parser. The second component contains aspect information that combines with the tense and mood on

[14] The paucal numbering is for definite plural reference, when the referents are present during the utterance of the sentence. It would be used in the translation of "Those several men [pointing] are whittling boomerangs." The plural numbering is for indefinite plural reference. This would be used in the translation of "Children play by the water hole."

[15] See [Nas86] for a complete table of tense elements.

the verb. The last component contains nominal agreement information that combines with the subject and object of the verb. (20) shows an auxiliary indicating that the aspect is imperfect, that the subject is first-person singular, and that the object is third-person singular in the dative case.[16] This form of the imperfective, *ka*, may be used only with the non-past tense.

(20) *ka-rna-rla*
IMPERF-1s-3d
imperfective aspect with first-person singular subject and dative object

When an element of the auxiliary is not uttered it does not vanish from the sentence. Instead, there is a default meaning for each of the components. As mentioned above, the default value for the complementizer position is null. This contrasts with other, overt complementizers that can appear, such as the negative that has scope over the entire sentence. When the base is not uttered, the auxiliary is given a perfective aspect, which combines with the tense on the verb, as with overt bases. The nominal agreement clitics default to third-person, singular. The null auxiliary, given in (21), is interpreted as containing perfective aspect and third-person, singular agreement.

(21) ϕ
perfective aspect with third-person, singular subject and object

1.1.2 Case Phrases

Before discussing case phrases, a bit more Warlpiri phonology must be introduced. As far as concerns the parser there are four levels of phonological grouping present within a sentence. The lowest level consists of morphemes, the indivisible units of the input.[17] The next level contains words which are sequences of morphemes. Not demonstrated explicitly in the examples so far is the third level, phonological phrases. These phrases are sequences of words, and are identified in a straightforward manner with word stress information.[18] In the examples given in this thesis, unless noted otherwise, each word corresponds to a single phonological phrase. The last level is that of sentences; the parser works on only one of these at a time.

Like English, noun phrases are not limited to a single noun. In Warlpiri, such phrases—actually called *case phrases* as they are identified by their case-marking—may consist of several nouns within a single phonological phrase.[19]

---

[16] The nominal agreement clitics are labeled in two parts. The first gives person and number information. For example, '1' stands for first-person singular. The second part is 's' for subjects and 'o' for objects. Dative objects are marked with 'd.'

[17] Morphemes are not part of phonology proper, rather of morphology. The basic units of phonology are really phonemes. I have assumed some morphological processing which combines the phonemes into morphologically indivisible units.

[18] In Warlpiri, the left-most word of a phrase receives primary stress and the remainder receives secondary stress. Thus, it is a simple operation to delimit the extent of phrases based on the stress information present in the surface string.

[19] There are also discontinuous case phrases where different parts of the case phrase appear in phonological phrases separated by other phrases. This phenomenon is not yet handled by the parser, however.

The ordering of nouns within a phonological phrase is constrained, mostly due to case-marking considerations. Otherwise unmarked nouns may be marked for case by case-marked nouns to their right within the same phrase; the case-marker in this case has an extended scope over all of the nouns in the phrase, and not just over the noun to which it is enclitic. (22) gives an example of this phenomenon. *Marlu* is not marked for absolutive case because it appears in the phrase along with a case-marked noun, *i.e.*, *wiri-ki*. It would be ungrammatical for the case-marked noun to appear to the left of the unmarked noun; this is shown in (23).

(22) *marlu wiri-ki*
kangaroo big-DAT
'to/from the big kangaroo'

(23) **wiri-ki marlu*
big-DAT kangaroo

The syntax of continuous case phrases is actually a bit more complicated than discussed so far. There may be case-marked nouns appearing before unmarked ones, so long as the latter nouns are marked by a case-marked noun to their right.[20] (24) gives an example. The first dative case-marker, *-ku*, has scope over the first word, *marlu*. The second dative case-marker, *-ki*,[21] has scope over the second two words, *pukurlpa* and *wiri*. The second word, *pukurlpa*, is allowed here because there is a case marker to its right, *i.e.*, *-ki*.

(24) *marlu-ku pukurlpa wiri-ki*
kangaroo-DAT friendly big-DAT
'to/from the big, friendly kangaroo'

1.1.3 Agreement

The auxiliary contains two major components, the base and the nominal agreement clitics. These components must agree with other parts of the sentence in which the auxiliary appears. There are tense restrictions on the base which must agree with the tense contained in the tense element enclitic to the verb stem. The more common bases are given in table 1.2.[22]

The first sentence, (1), provided an example of grammatical agreement of the auxiliary base and the tense element on the verb. The base in this instance was *ka*, that requires the tense on the verb, *rni*, to be non-past, which it is. The same sentence uttered with the other imperfective base, *-lpa*, would not be grammatical, as there would be a tense clash. See (25).[23]

---

[20] The parser does not handle this type of case phrase, only the simpler form where there are some number of unmarked nouns followed by a case-marked noun which has scope over the entire phrase.

[21] *Ku* and *ki* are allomorphs of the dative case-marker.

[22] A complete list of the auxiliary bases can be found in [Nas86].

[23] *-Lpa* is enclitic to the preceding word, *ngajulu-rlu*, because it is a clitic, not capable of beginning a word on its own.

base	aspect	tense restrictions
ϕ	perfective	(none)
ka	imperfective	non-past
lpa	imperfective	past, irrealis

Table 1.2: Common auxiliary base clitics.

(25) **Ngajulu-rlu-lpa-rna-rla punta-rni kurdu-ku karli.*
I-ERG-IMPERF-1s-3d take-NONPAST child-DAT boomerang

The other component of the auxiliary is the nominal agreement clitics. They contain person and number information which must agree with the person and number information of the argument case phrases. When no number-marker is present on a noun it may either be interpreted as singular or plural; the information in the matching agreement clitic determines which. This distinction is shown in (26). In (a) the subject, *wati*, is singular because the subject agreement clitic is null, denoting third-person, singular. In (b) the subject is plural because the subject agreement clitic, *lu*, denotes third-person, plural.

(26) a. *Wati-ngki-palangu pantu-rnu marlu-jarra.*
man-ERG-33o spear-PAST kangaroo-DUAL
'The man speared the two kangaroos.'
b. *Wati-ngki-li-palangu pantu-rnu marlu-jarra.*
man-ERG-333s-33o spear-PAST kangaroo-DUAL
'The (several) men speared the two kangaroos.'

1.1.4 Auxiliary Positioning

The auxiliary word must appear either at the beginning of a sentence or in Wackernagel's position[Wac92], the "second" position. More precisely, the second position occurs at the end of the first phonological phrase of the sentence or in the second phrase by itself. The auxiliary may either be a word unto itself or appear as a clitic on the last word of the phrase.

There are more constraints on the positioning of the auxiliary. In Warlpiri, words must have two or more syllables. Therefore, if the auxiliary has only one overt syllable, *e.g.*, *ka*, then it must be enclitic; hence it must appear in Wackernagel's position, rather than at the beginning of a sentence. There is one exception which is the auxiliary base, *lpa*, which is a clitic, and may not begin a word even if it begins an auxiliary with two or more syllables. Note that the agreement markers are also clitics and also may not begin a word (*i.e.*, in the event of a phonologically null base).

The sentences in (27) demonstrate grammatical placements of the auxiliary. In (a) the left-most element of the auxiliary, *rna-rla*, is an agreement marker and

therefore it must be enclitic to the preceding word. In (b), the auxiliary consists of a single syllable, so it too must be enclitic and in Wackernagel's position. (c) demonstrates another auxiliary in second position; note that the first phrase consists of two words. In (d) an auxiliary in first position is shown.

(27) a. *Ngajulu-rlu-rna-rla punta-rni kurdu-ku karli.*
I-ERG-1s-3d take-NONPAST child-DAT boomerang
'I will take the boomerang from the child.'

b. *Kurdu-ka nya-nyi wati-ngki.*
child-IMPERF see-NONPAST man-ERG
'The man sees the child.'

c. *Marlu wiri-ki-rna-rla karli punta-rni ngajulu-rlu.*
kangaroo big-DAT-1s-3d boomerang take-NONPAST I-ERG
'I will take the boomerang from the big kangaroo.'

d. *Kalaka-npa-rla karli kurdu-ku punta-rni.*
ADMON-2s-3d boomerang child-DAT take-NONPAST
'You might take the boomerang from the child.'

The sentences in (28) are not grammatical. In (a) the auxiliary, *rna-rla*, appears enclitic to the word in the second phrase. In (b) the clitic, *rla*, begins a word. And in (c) the auxiliary appears cliticized to the word in the third phrase; the fact that it is enclitic to the verb makes no difference.

(28) a. **Ngajulu-rlu punta-rni-rna-rla kurdu-ku karli.*
I-ERG take-NONPAST-1s-3d child-DAT boomerang

b. **rla kurdu-ku punta-rni karli wati-ngki.*
3d child-DAT take-NONPAST boomerang man-ERG

c. **Marlu wiri-ki karli punta-rni-rna-rla ngajulu-rlu.*
kangaroo big-DAT boomerang take-NONPAST-1s-3d I-ERG

1.1.5 Argument Identification

There is an important relation between the verb and the case phrases in a simple sentence, namely, the relation of predication. That is, the verb acts like a logical predicator, taking the case phrases as its arguments. This relation is manifested in two ways. Syntactically, case phrases may appear as the subject of a sentence, as well as a direct object and indirect object. This is distinct from the semantic use of case phrases in which they are identified with the different roles which the verb selects. Let's clarify this two-level analysis by considering (1) once again. In this sentence, the ergatively marked pronoun, *ngajulu*, takes on the subject function; *karli*, the noun marked for absolutive case, takes on the object function; and *kurdu*, marked for dative case, takes on the indirect object function. From the semantic point of view we see that *ngajulu* is the taker, that *karli* is the thing taken, and that *kurdu* is the source from which the object is taken. It is important that the parser be able to determine this mapping from case phrases to arguments, shown in (29), as part of the meaning of the sentence.

(29) taker ↔ *ngajulu* ('I')
taken ↔ *karli* ('boomerang')
source ↔ *kurdu* ('child')

1.1.6 Null Anaphora

The last phenomenon to be covered is known as "null anaphora." In Warlpiri, case phrase arguments need not appear overtly in a sentence. When this happens, the referent of the missing argument is retrieved from context. Suppose, for example, that the speaker had been talking about his son when he uttered (30), which is the same as (1) with *kurdu-ku* missing.

(30) *Ngajulu-rlu ka-rna-rla punta-rni karli.*
I-ERG IMPERF-1s-3d take-NONPAST boomerang
'I am taking the boomerang from him/her/it.'

This sentence would be understood as referring to the speaker's son, as in "I am taking the boomerang from my son." Note that not any referent may be used because it must still register with the agreement clitic in the auxiliary, in this case third-person singular.[24]

This section has presented the phenomena that the parser can handle. However, it remains to specify the parser itself, beyond the very brief overview given earlier. In order to understand how the parser is situated in the science of natural language processing, it will first be necessary to outline the methodology of the research.

1.2 The Methodology of Natural Computation

The model of parsing proposed in this thesis falls into the theoretical framework of natural computation. In this approach there are four components: the abstract computational theory, the representation and algorithm, the implementation, and the test. The first three elements of this synthetic methodology are described by Marr[Mar82]:

> At one extreme, the top level, is the abstract computational theory of the device, in which the performance of the device is characterized as a mapping from one kind of information to another, the abstract properties of this mapping are defined precisely, and its appropriateness and adequacy for the task at hand are demonstrated. In the center is the choice of representation for the input and output and the algorithm to be used to transform one into the other. And at the other extreme are the details of how the algorithm and representation are realized physically -- the detailed computer architecture, so to speak. [pp. 24-5]

[24]Since the parser handles but one sentence at a time, checking agreement with null anaphora is not actually performed. Instead, the parser simply allows for non-overt arguments.

The last element of the natural computation approach, testing the implementation, is necessary to provide corrective feedback for the first three elements. In addition to providing the abstract computational theory, the representation and algorithm, and the implementation, one must argue that they are faithful to one another—that the algorithm, in fact, computes the mapping of the computational theory, and that the implementation is a correct realization of the algorithm. Once this is done, the test of the implementation can be said to be a proper test of all three components, especially the computational theory.

How can the methodology of natural computation be applied to the problem at hand? As for the first part, GB will be used as the computational theory. GB defines the mappings between the sentence and the syntactic structures underlying it. Because the aim is to build a parser, we will be computing the mappings in the direction from sentence to structure. The next step of the solution, then, is to design the algorithm and representation that compute the mappings. After this the design must be implemented, and, finally, tested on a natural language, Warlpiri.

As noted above, it is necessary to produce algorithms and representations that are faithful to the computational theory. To this end, I will employ the *type transparency* hypothesis, as described in [BW84]:

> ... the condition that the logical organization of the rules and structures incorporated in a grammar be mirrored rather exactly in the organization of the parsing algorithm. [p. 39]

Of course, not all algorithms need be constructed so directly from the computational theory. This hypothesis is appealing because it minimizes the argumentation needed in order to show that the algorithm is faithful. (Additional, independent support for the hypothesis is given in [BW84].) By showing that the algorithm and implementation mirror the grammar as defined by GB closely, I hope to show that the solution properly answers the questions put forth in the thesis.

1.3 The Abstract Computational Theory

This section presents a brief description of the parser at the level of abstract computational theory. The theory is fully presented in the following chapter.

The task of the abstract computational theory of the parser is to specify the mapping of the input sentence to the output structures. The parser assumes that some processing of the input sentence (*i.e.*, speech stream) has been performed. The input to the parser can be characterized as a four-tiered structure. At the top level is the sentence to be parsed. Sentences consist of a number of phonological phrases; phrases consist of a number of words; and words consist of a number of morphemes. As an example, (1) is given below in the input representation:

```
(31)  (((NGAJULU RLU) (KA RNA RLA)) ((PUNTA RNI)) ((KURDU KU))
      ((KARLI)))
```

The structure of the output is given by two linguistic theories, GB and a theory of the lexicon.[25] One of the goals of GB is to account for linguistic phenomena (mostly syntactic) with a number of levels of representation. Each of these levels is concerned with a certain aspect of the linguistic information contained in a sentence. The idea is that each level represents only what it needs to in order to account for that aspect with which it is concerned; other levels represent the information appropriate to their domains.

This approach differs substantially from earlier formulations of natural grammars.[26] These grammars consisted of a single set of rules used to generate structures corresponding to surface strings. A string was said to be grammatical if it could be generated by the grammar, and ungrammatical if not. The GB-style approach, on the other hand, uses several structures to generate surface strings, but each is concerned only with some aspect of the sentence. Because more than one structure is used, a sentence may be partially grammatical. That is, it may be grammatical with respect to some aspects of the grammar, and ungrammatical with respect to others. This formulation has an intuitive appeal. Consider (32). Formally speaking, this sentence is ungrammatical, yet it is understandable. If only *went* and *I* were interchanged, the sentence would be completely grammatical. Roughly speaking, GB would represent the partial grammaticality by stating that (32) has an ungrammatical precedence structure, but a grammatical syntactic one (*i.e.*, that case-marking conditions are violated). The grammatical syntactic structure allows semantic interpretation.

(32) Went I to the store with Mary.

The version of GB adopted in this thesis comes largely from the mainstream work in the field. However, there are many variations of the theory extant, mostly due to the youth of the endeavor. As a result, it draws from some of the GB work specifically focused on Warlpiri and similar languages. The major differences here concern the formulation of the subject grammatical relation, and use of the basic category AUX instead of the more common INFL (inflection). The following chapter on the linguistic theory will point out where the two accounts diverge.

In the development of the parser I have had to make a further modification to the underlying theory to account for free word order. There have been proposals in the literature,[27] but none seemed to work out and still maintain a concordance with GB. The change adopted in this thesis is that S-structure and the other syntactic structures are unordered, leaving them solely as hierarchical entities. This too will be elaborated in the following chapter.

GB is composed of three main levels of representation, phonological, logical, and syntactic. The phonological level is meant to capture the sound structure of utterances, while the syntactic level represents the syntactic relations among the constituents that the phonological level has highlighted. The logical level, while

[25] There are many theories that come under the purview of lexical theory, and there are correspondingly many sources. A good choice is [Lev85] and the references mentioned within.

[26] [Cho65] is a major work of this era of transformational linguistics.

[27] See [Nas86] and the discussion of other theories there.

a necessary part of a more nearly complete grammar and highly developed in GB theory, awaits an instantiation in a future version of the parser. The parser computes some of the phonological and syntactic levels of representation. A graphic depiction of the processing is given in figure 1.8.

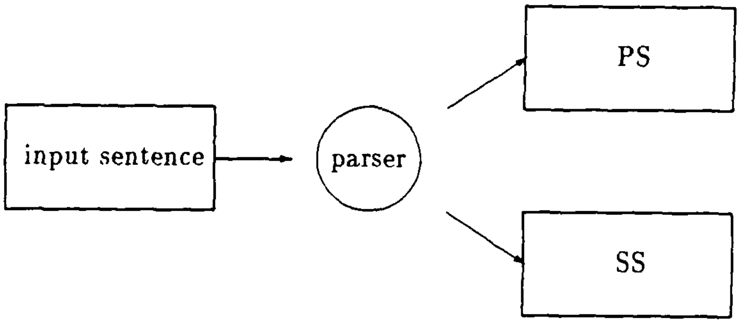


Figure 1.8: The parsing process.

In GB the level of phonological structure is called "Phonological Form" (PF). PF represents many aspects of a sentence, such as pitch, stress, and meter. Only a part of PF, namely, precedence and adjacency relations, are used to form the basis of parser's precedence structure. PS represents this information for all levels of the input, morphemes, words, and phrases. By representing the precedence and adjacency for the morphemes of a word, PS also encodes some morphology; yet, it cannot be called a morphological structure, as it also deals with units larger than a word.

The phenomena accounted for by PS are given in table 1.3.[28] The composition of nouns, verbs, and auxiliaries is covered by ordering among morphemes. Word order is used to account for continuous case phrases. Finally, auxiliary positioning is concerned with ordering of a word (or clitic) among the phrases of a sentence, specifically, the first or second phrase.

- nominal, verbal, and auxiliary composition
- continuous case phrases
- auxiliary positioning

Table 1.3: The phenomena represented in PS.

The syntactic level is further broken down into two sublevels, S-structure and D-structure, that are meant to capture the syntactic regularities that appear at both a superficial and deep level of analysis. S-structure corresponds more closely to the

[28] Of course, the parser handles but a small subset of Warlpiri. The concluding chapter discusses the phenomena that remain for future versions of the parser.

surface utterance, making explicit the syntactic relations of its components, such as verbs, nouns, and the larger constituents, verb and noun phrases. D-structure represents the canonical form of the sentence, which may have several surface manifestations. A quick example will bring out the distinction. Consider the active and passive forms of a sentence, such as in (33). These sentences appear quite different when spoken (or written), but they seem to say the same thing. The difference in appearance would be captured in differing S-structures for each of the sentences. However, the similarity would be captured with identical D-structures for each sentence.

(33) a. I took the boomerang from the child.
b. The boomerang was taken from the child by me.

S-structure and D-structure are related by the single relation of *movement.* That is, elements of D-structure may move from their original positions to other positions in the structure. This will be elaborated in the following chapter. What should be noted here is that for the parser there is no difference between S-structure and D-structure because no movement is necessary in simple Warlpiri sentences. For this reason, the parser need only compute one syntactic structure (SS), and not two. In more traditional GB theory, S- and D-structure represent precedence as well as hierarchical information. Thus, movement is needed to account for either movement in the surface string (precedence), or movement in the syntactic structure (hierarchy), or both. Because the syntactic structures adopted for the parser are not ordered by precedence, no movement is necessary for permutation in the surface string. The simple range of phenomena covered by the parser demand no hierarchical movement, so S- and D-structures collapse. This doesn't constitute an argument against two levels of syntactic representation, as they seem to be necessary cross-linguistically. However, the parser need only represent one level.

GB consists of a number of other subtheories, each of which defines the relations that may obtain at each of the levels, and the constraints which grammatical structures must satisfy. $\overline{\text{X}}$-theory, for example, defines the structural possibilities in syntax; it gives the basic possible structures for verb phrases, noun phrases, and so on. The theory of government defines a widely used structural relation, government, that seems to pervade the analysis at both syntactic levels. Case theory covers the usage of case, as for example, in the case-marked noun, *kurdu-ku.* The last subtheory used by the parser, θ-theory, is about the semantic subcategorizations of predicators. The verb stem, *punta*, for instance, subcategorizes for three semantic arguments, one for the taker, one for the object taken, and for the source of the taking.

Syntactic structure (SS) accounts for a number of phenomena, as given in table 1.4. Free ordering of phrases is represented in SS, mostly because there are no ordering constraints among phrases in PS: SS is unable to impose any such constraints as precedence is not represented at that level.[29] As claimed in the theory,

[29] It is believed that discontinuous case phrases will also represented in SS. To handle these phrases, the continuous case phrases with similar case-marking would be adjoined to the same argument position in the syntactic structure.

grammatical functions are solely hierarchical relations, so they too are represented in SS.

- free phrase order
- grammatical functions

Table 1.4: The phenomena represented in SS.

The parser is also based on a limited semantic theory. The semantics extends as far as interpretation of the syntactic structures is possible.[30] The parser covers four kinds of semantic interpretation, listed in table 1.5. Argument identification is the process of relating case phrases to their argument positions of predicates, which, for the sentences in the domain of the parser, are verbs. Null anaphora occur when there is no overt argument in the sentence, yet there is an understood argument, usually gleaned from discourse context. Because the parser processes one sentence at a time, the argument is only left flagged as referring to something outside the sentence. Null auxiliary components are interpreted with their default values, as described above. The last phenomenon, auxiliary agreement, is rather straightforward. The aspect information is combined with the tense and mood on the verb's tense element, subject to the tense restrictions on the base; the nominal agreement information from the auxiliary is combined with the arguments (*i.e.*, case phrases) in the sentence.

- argument identification
- null anaphora
- null auxiliary components
- tense and argument agreement

Table 1.5: The phenomena interpreted by semantics.

Lexical theory provides an account of the information associated with each lexical item.[31] Each item maps to a lexical entry that contains its category (*i.e.*, part of speech), information for its role in both precedence and syntactic structures, and semantic information. This is illustrated in figure 1.9. The information in the item's entry is what determines its interaction in the structures at each of the levels. In PS, for example, the item's cliticization information determines what entities it can be enclitic to. The precedence part of the entry for case-markers, for instance, indicates

[30] In GB semantic interpretation is actually performed on the level of Logical Form (LF), another level of syntax. As with D-structure, LF is related to S-structure by movement. The simple part of Warlpiri that is covered by the parser does not call for any movement in the mapping to LF, so SS may serve for semantic interpretation as well.

[31] This version of the parser does not handle lexical ambiguity, so each lexical item maps to a unique lexical entry.

that they may be enclitic to nouns; nouns, on the other hand, may not be enclitic to other items.

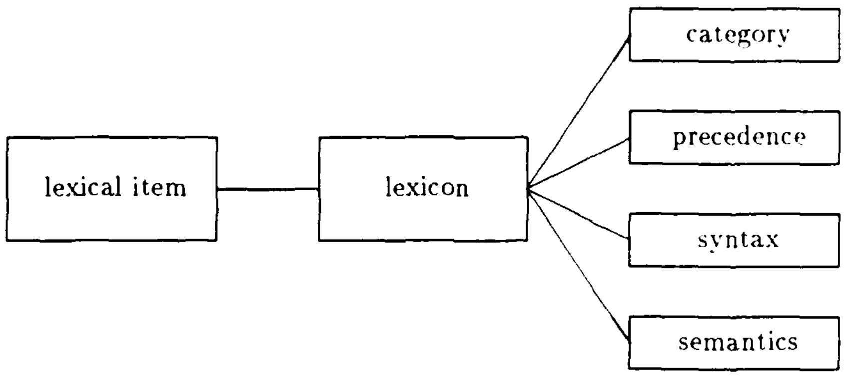


Figure 1.9: The mapping of the lexicon.

The way in which lexical items show up in syntactic structure is more complicated than precedence structure. In PS, each lexical item is entered as a single node in the structure, where it is subject to the rules of combination of precedence structure (*e.g.*, cliticization). In SS, however, each lexical item *projects* as a lexical structure (L-structure) that represents the part of the sentential syntax that corresponds to that item. The form of the L-structure depends on the meaning of the lexical item. The theory of lexical semantics adopted for the parser divides lexical items into two semantic classes, predicators (*e.g.*, verbs) and arguments. The L-structure for an argument is simply a node. The L-structure of a predicator, on the other hand, is a structure that has positions in it for arguments. For example, consider (1) once again. In this sentence there are several lexical items. The nouns, since they are not predicative, have L-structures that are simply nodes.[32] The verb, in contrast, is predicative, and thus maps into an L-structure containing positions for its arguments. These L-structures (and those of the other parts of the sentence, of course) are combined by rules of syntax to form D-structure. This will be elaborated in the following chapter.

The semantic aspect of a lexical entry is, perhaps, its heart. One of the major hypotheses of lexical semantics is that the meaning of an item in large part fixes its syntactic manifestation. While the work is still ongoing, a number of confirming results have appeared. *The parser adopts one of these theories, concerning the* syntactic manifestation of arguments for predicators. That is, the argument positions in the L-structure of a predicator need not be specified in the syntactic part of the entry; they are, instead, determinable from the semantic information which encodes a list of the arguments and their types. There is a set of rules executed by

[32] GB provides a fuller account of nouns phrases, giving them a projected L-structure due to their predicative nature. In this thesis, however, nouns are taken to be unanalyzable entities. A more robust version of the parser will have to allow for predicative NPs.

the parser that will map semantic arguments to the grammatical functions in which they participate in the syntactic structure. (34) contains some sample rules.

(34)
1. The agent of the action appears as the subject.
2. The patient of the action appears as the object.
3. The source of the action appears as the indirect object.

Consider (1) one more time. The main predicator is the verb stem, *punta*. In its lexical entry is the semantic information that it takes three arguments, a taker, a taken, and a source from which the taken is taken. The taker is the agent of the action, and what is taken is the patient. Applying the rules in (34), the parser determines that the taker will appear as subject, and so on. This was shown in the mapping of grammatical functions to words that fill the argument positions in (3).

1.4 Coming Attractions

The remaining levels of description need no introduction, and it is best to simply present them straightaway. The following chapter provides a complete description of the linguistic theory underlying the parser. The design of the parser is given in the chapter, Representation and Algorithm. Because the implementation follows the design quite closely, no separate chapter is needed to discuss it. Instead, a few implementation notes are given in the appendix. The appendix also contains a battery of grammatical and ungrammatical inputs given to the parser to demonstrate its coverage of the advertised phenomena. The thesis concludes with an evaluation of the shortcomings of the parser, and a comparison of the parser with similar work in the field.

Chapter 2

The Linguistic Theory

Government-Binding theory (GB) and lexical theory comprise the representational foundation of the Warlpiri parser. These theories are by no means complete or well-understood, however. As with other scientific theories, they are in a constant state of flux, changing rapidly as new insights are made. In this chapter I will state the particular formulations of these theories that the parser assumes.

Both GB and lexical theory come under the rubric of generative linguistic theory. They can be viewed as an intrinsic specification of the grammatical sentences of a language, much like a logical predicate which implicitly denotes the members of the set of elements for which it would yield true. However, not any statement of the grammar will do. The power of GB and the lexical theory is their modularity and regularity that give them an explanatory punch. The theories consist of a small number of components that combine to make powerful predictions about grammaticality.

In an attempt to explain the universal aspects of grammar (*i.e.*, the features common to all languages), the modules of these theories are stated in a general manner. Even though languages of the world seem to exhibit similar phenomena. they do so in differing ways. So a single theoretical statement can not suffice to account for the varying data. On the other hand, a theory that lists each of the cases serves no explanatory function. GB resolves this discrepancy with the notion of *parameterization*. The theories within GB are formulated in general way, yet they are subject to limited parameterization for particular languages. An example can be found with the phenomenon of agreement. In Warlpiri, both the subject and the object have agreement markers in the auxiliary. In English, however, there is only subject agreement (found on the tense marking of the verb). In general, one might state that the arguments of a verb must agree with the agreement markers. The parameter for Warlpiri would state that both the subject and object are involved, while for English the parameter would be set to subject only.

But these theories do more than merely determine membership of a sentence in a language. They also impart linguistically relevant structures to the sentences that make the syntactic information contained in them explicit. One such structure is the relationship between a predicator[1] and its arguments. That is, the theories

[1] By 'predicator' I mean an entity that takes arguments, like a logical function. One syntactic

identify the predicator and its arguments in the sentence, and then determine the relationships between them. In simple sentences, this means ascertaining which word corresponds to the verb and which phrases (*e.g.*, case phrases in Warlpiri) correspond to its syntactic arguments.

The first component of the parser's linguistic basis, Government-Binding theory, contains several levels of representation, concerning the phonological, syntactic, and logical aspects of a sentence. This is shown in figure 2.1. Each of these levels can be thought of as a different view on the sentence to which they correspond. Looking through "syntactic sunglasses" each of these levels filters out the information in the sentence that does not apply to itself, letting only the pertinent information through.

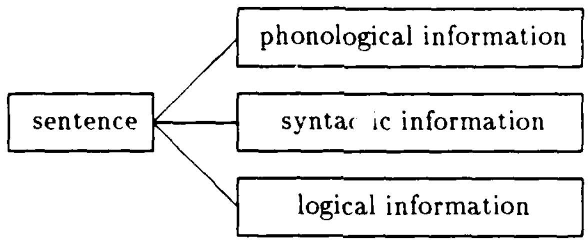


Figure 2.1: The principal levels of GB.

Figure 2.2 shows the GB model of grammar assumed by the parser. The level of logical representation is not shown because the parser does not compute logical structures. While the level of "Logical Form" (LF) is an important component of sentence meaning, it has not yet been dealt with in the parser.

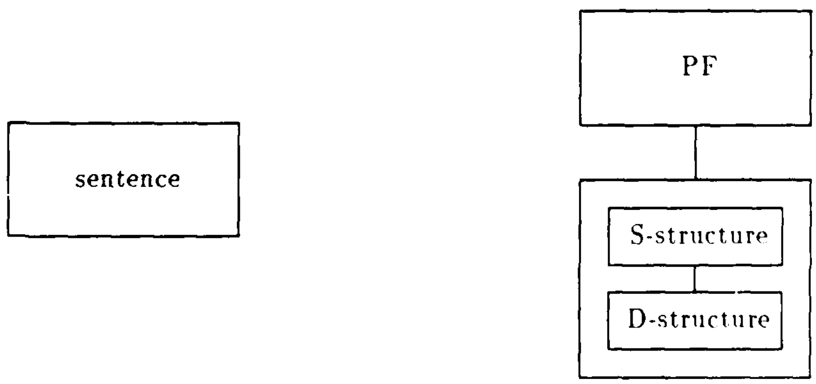


Figure 2.2: The GB model of grammar.

instance of a predicator is the verb which may take subjects, objects, and indirect objects as arguments.

The level of "Phonological Form" (PF) represents the phonological aspects of a sentence, such as pitch, stress, and meter. However, only a part of PF is used here. As stated in the introduction, the parser assumes instead a level of precedence structure (PS) that represents both precedence and adjacency relations, as found in the traditional level of PF. Note that PS also incorporates some morphology, as it represents the precedence and adjacency of the morphemes within a word.

The domain of PS consists of morphemes, words, and phrases, as all of these elements may be involved in precedence relations. In Warlpiri, as it turns out, there are no precedence constraints between phrases, so only morphemes and words will be represented in PS. In other languages, such as English, precedence among phrases is important, so they will be manifest in PS.

The syntactic component is the heart of GB as it stands today. As GB is a transformational theory, the syntactic level is composed of two parts, the base and a set of transformations. The base is the set of structures that correspond to the canonical form of sentences; this is represented in D-structure. The set of transformations can be applied to D-structure to yield surface structures (S-structures) that correspond to surface sentences, the sentences that we actually utter. This format offers a perspicuous representation for capturing both the similarity and disparity of different syntactic constructs. One example of this phenomenon was mentioned in the introduction, namely, pairs of sentences in active and passive voice. GB analyzes these sentences as having similar D-structures since they seem to have the same structure at a deeper level. The alternation in surface form is reflected in the differing S-structures that GB assigns to the different voices.

GB claims that there is a connection between PS and the level of syntax; that is, that there are conditions that impose mutual constraints between the levels. This is easily demonstrated. Consider the sentences in (1). The relation between precedence and syntax is systematic: the subject is the first noun phrase, and the object is the second.

(1) a. John likes Mary.
    b. Mary likes John.

As far as the parser is concerned the connection between PS and syntax exists exactly where the surface order of constituents has an effect on the syntactic analysis. For the subset of Warlpiri covered, this concerns only the relation between a case-marker and the nouns over which it has scope (*i.e.*, those nouns to its left within the phonological phrase). In Warlpiri, case-markers must be suffixed to nouns. When a noun and a case-marker are in such a configuration, the noun is identified as a syntactic argument of the case-marker. Conversely, the nominal argument of a case-marker will appear in PS as a noun with the case-marker suffixed onto it, and in no other way. In this instance the precedence and syntactic structures are in a highly constrained, one-to-one relationship.

At this point I must repeat a caveat mentioned in the introduction. The particular formulation of GB presented here is actually an amalgam of three sources. Mostly, it comes from mainstream GB, but there are some parts that find their roots in the literature of Warlpiri linguistics. The third source is the set of modifications

to GB theory found in this thesis. The main contribution here is the removal of precedence from syntactic structures. Another contribution is the formulation of precedence structure, which borrows from two more traditional representations of GB, PF and morphology. I will point out the differences from mainstream GB as they arise.

The second part of the linguistic foundation, lexical theory, adds two more components to the model of grammar assumed by the parser, as shown in figure 2.3. The lexicon is the mapping between morphemes and PS and syntax. Associated with each morpheme is precedence and syntax information. The precedence information determines how the morpheme is manifest in PS; similarly, the syntactic information determines its syntactic manifestation. Case-markers, for example, must be enclitic (*i.e.*, affixed) to nouns at the level of PS, but at the level of syntax there is no such requirement, as cliticization is not relevant at that level. Instead, the case-marker is the head (central element) of its phrase, taking nouns as its arguments.

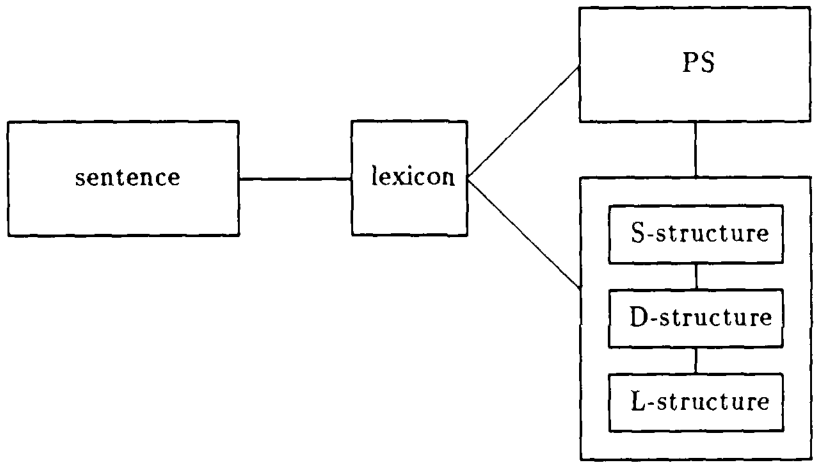


Figure 2.3: The parser's model of grammar.

Syntactic manifestation of lexical items is carried out via lexical structures (L-structures). L-structures are the syntactic structures that correspond to a single lexical item. L-structures are combined to produce the D-structure for the entire sentence. For example, consider a sentence with a transitive verb. Roughly speaking, there are three lexical items, namely, the verb, and its two noun arguments. Each of these items is manifest in syntax as an L-structure. Their L-structures are combined with syntactic relations to form the D-structure for the entire sentence.

Before finishing this overview of the parser's grammar, one more point needs to be made, concerning the form of a sentence. The basic units of analysis are taken to be morphemes, rather than, say, more elemental units such as phonemes. It is further assumed that sentences consist of several levels: words, which are sequences of morphemes; phrases, which are sequences of phrases; and, sentences, which are sequences of phrases.

The remainder of the chapter will describe the model of grammar, depicted in figure 2.3, in greater detail. First, the level of PS will be presented. After this is the section on syntax, that will cover each of the sublevels, L-structure, D-structure, and S-structure. The following section discusses the theory of the lexicon. The chapter concludes with a description of the theory of semantic interpretation used by the parser.

2.1 Precedence Structure

This section begins with a description of precedence structure. The theory will then be applied to the phenomena to be accounted for with this structure; see table 2.1. The theory works straightforwardly for both nominal and verbal composition, and continuous case phrases. However, the auxiliary is a strange entity, and is not captured so neatly. This section ends with the extensions to the theory necessary to handle auxiliary composition and placement.

- nominal and verbal composition
- continuous case phrases
- auxiliary composition
- auxiliary positioning

Table 2.1: The phenomena accounted for in PS.

The elemental units of PS, morphemes, are combined by rules into larger structures. PS is recursive in that the resulting structures may in turn be combined by these same rules (an example will be given below). The rules are constrained to operate only on adjacent elements of the structure, ordered by precedence.

Each of the elemental units is labeled with its category, as categorial information is needed in PS. Consider the sample sentence from the introduction, repeated here as (2). The PS for this sentence before having applied any rules is shown in figure 2.4.[2]

(2) *Ngajulu-rlu ka-rna-rla punta-rni kurdu-ku karli.*
I-ERG IMPERF-1s-3o take-NONPAST child-DAT boomerang
'I am taking the boomerang from the child.'

The basic rule for PS is combination, given in (3).[3] This rule allows any node to combine with any other node. However, there are empirical restrictions on combination. These restrictions are captured by four interacting parameters. The first

[2] The labels for nouns, case-markers, verb stems, and tense elements are 'N', 'C', 'V', and 'T', respectively. The auxiliary consists of four optional components, the base, the subject and object agreement clitics, and the dative registration clitic, labeled, 'B', 'S', 'O', and 'D', respectively.

[3] The ideas for the rule of combination and the parameters of variation are taken from chapters 2 and 5 of [Nas86]. Nash uses the notion of a *categorial signature* to represent the constraints of combination. The description presented here is largely a reformulation of his theory.

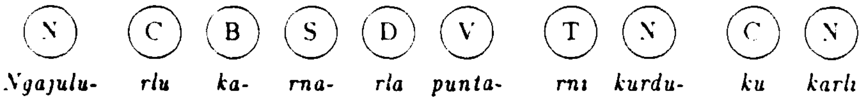


Figure 2.4: The elemental units of the PS for (2).

parameter concerns the direction of combination; one node acts as the combiner and the other acts as the combinee. The direction of combination is invariant across the language; for Warlpiri, the direction is from right to left.

(3) Combine two adjacent nodes.

The other three parameters of variation depend on the category of the combiner, and therefore the parameter settings are stored in the lexicon on a per category basis. The second parameter concerns the categorial restrictions of combination. That is, some categories may combine with some categories and not with others. The third parameter covers the assignment of category to the root of the newly created structure. The choice is restricted to either the category of the left node or the right node. The last parameter of variation dictates the phonological level (*i.e.*, word, phrase, or sentence) at which the combination takes place.

We can now account for three of the Warlpiri phenomena listed above: nominal and verbal composition, and continuous case phrases. The PS for declined nouns is demonstrated with the noun, *kurdu-ku*. There are two elemental nodes in the PS for this word, one for the noun, and one for the case-marker. The rule of combination may apply because the combinee, *kurdu*, fits the parameters settings of the combiner, *ku*. First of all, the combinee is to the combiner's left, as required by the language-wide direction parameter. Secondly, *ku*, being a case-marker, may combine with nouns at the word level. Finally, we see that the new root is given the case-marker category. The resulting PS is shown in figure 2.5.

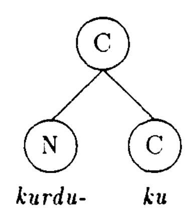


Figure 2.5: The PS for *kurdu-ku*.

Nouns marked for absolutive case are covered with a special rule of PS. In the event of a non-overtly marked noun, PS supplies a null category, the absolutive case-marker. This category must be posited because the absolutive case, like the other

syntactic cases, marks the noun to which it is enclitic, and the preceding nouns in the phonological phrase (see below). The PS for the absolutive argument of the sample sentence, *karli*, is depicted in figure 2.6.

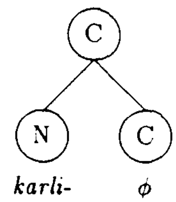


Figure 2.6: The PS for *karli*.

The PS for inflected verbs is also accounted for by the parameterized rule. The difference here concerns the combiner, which is the tense element. It combines with verb stems rather than nouns, and the resulting category of the combination is verb instead of tense element. The difference in transmission of category is due to syntactic effects, described below. The PS for the verb, *punta-rni*, is given in figure 2.7.

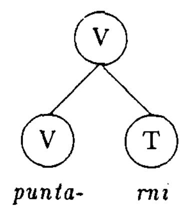


Figure 2.7: The PS for *punta-rni*.

The last phenomenon handled by the rule of combination is that of continuous case phrases. Consider the phrase in (4). In this phrase, the case-marker, *-rlu*, has scope over all three nouns. This analysis is grammatical because case-markers may also be adjacent to nouns at the phrase level (in addition to the word level, as with the declined noun, above). The PS for this phrase is shown in figure 2.8.

(4) *yirrinji yirraru kardirrpa-rlu*
centipede homesick brave-ERG
'the brave, homesick centipede'

There are two well-formedness conditions for PS. The first, given in (5), states that PS must not contain any uncombined structures. However, this condition is too severe for all languages. In Warlpiri, for instance, there are no ordering constraints between phrases, and so PS will not contain connected structures between

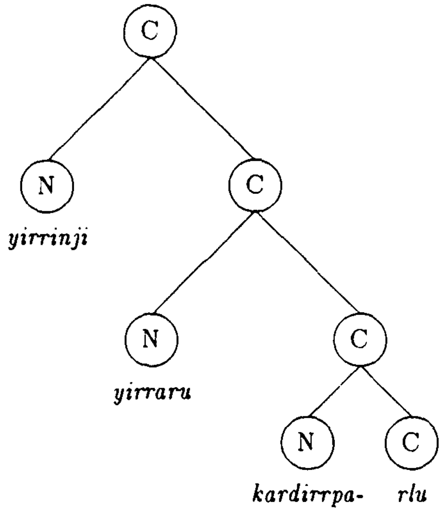


Figure 2.8: The PS for (4).

phrases. The facts are covered with a parameter of variation that dictates at which phonological levels the condition applies. This condition applies both at the word and the phrase level for Warlpiri, but not at the sentence level. Thus, the PS for words and phrases must contain single structures, but there is no such requirement at the sentence level.

(5) PS must be fully connected.

This single condition, in conjunction with the parameters of variation for the rule of combination, serves to rule out many types of ungrammatical words, phrases, and sentences that are ungrammatical with respect to precedence. (6) contains three types of ungrammatical words, all of which are ruled out by the well-formedness condition. (a) gives an example of a category mismatch: tense elements may not combine with nouns. (b) shows a word with morphemes in the wrong order: because verb stems do not combine with tense elements, the two elemental structures won't be combined into one, and the well-formedness condition will rule this word out. (c) shows the verb with the components in the right order, but in separate words. Because the tense element is constrained to combine at the word level, no combination takes place, and the condition rules this one out too.

(6) a. **kurdu-rni*
child-NONPAST
b. **rni-punta*
NONPAST-take
c. **punta rni*
take NONPAST

The second well-formedness condition concerns the composition of words in Warlpiri, and is stated in (7). Words that contain just one syllable are not grammatical, instead they must be enclitic to a preceding word. The auxiliary, *ka*, is an example of such a clitic. Note that all nouns and verbs automatically pass this condition, but for different reasons. Nouns pass because, as seems to be the case, there are no single-syllable noun stems.[4] Verbs pass because they must be inflected for tense: there are no null verbs nor tense elements.

(7) Words must consist of at least two syllables.

2.1.1 Auxiliary Composition

The auxiliary is an irregular word. Unfortunately, only a descriptive theory of its composition is available. It consists of a number of morphemes, all of which may or may not be present in the surface string.[5] The parser covers the part of the auxiliary word consisting of the base, the agreement clitics, and the dative registration marker. Their positioning within the word is best given by a template, as shown in figure 2.9.[6]

base	subject	object	dative

Figure 2.9: The auxiliary template.

The PS for the auxiliary is also built up with the rule of combination, however, two modifications are required. First, the categorial restrictions of the combiners must allow for the optionality of the elements. That is, the combiners may not have static categorial restrictions; instead, the restrictions must be dynamically determined, depending on the overt morpheme sequence. The template, above, is consulted to determine grammatical sequences.

The second change concerns the construction of the auxiliary structure. Rather than combining adjacent nodes into a binary tree, the auxiliary morphemes are combined as siblings, children of a single parent node. This linear structure is used to reflect the simple template that describes the possible combination of the component morphemes. As an example, the PS for *ka-rna-rla*, the auxiliary of (2), is given in figure 2.10.[7] Note that this auxiliary word contains three of the four possible morphemes, omitting the object agreement clitic.

[4] Of course, this is entirely an empirical point. If it turns out that single-syllable nouns exist, they too would be subject to this condition.

[5] See [Nas86] for a more detailed discussion of the auxiliary components.

[6] Other parts of the auxiliary, such as the complementizer, await a future implementation; however, their appearance in the auxiliary word can be accounted for by extending the template.

[7] The category dominating the auxiliary word is labeled "A" for "AUX." AUX is a discontinuous part of the INFL of traditional GB that contains AGR and TNS. It is not so clear how INFL is manifested in Warlpiri, with the tense and agreement information spread over the auxiliary and the verb, so "AUX" is used instead.

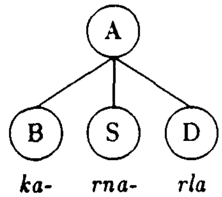


Figure 2.10: The PS for *ka-rna-rla.*

2.1.2 Auxiliary Positioning

The positioning of the auxiliary is quite unlike that of other words. Roughly speaking, auxiliary words may appear either in the first or second position of the sentence. This special property is accounted for in two ways. First, auxiliary words (with exceptions to be described below) do not combine with other words. That is, the precedence structures for auxiliaries are inert. Second, auxiliaries are considered to be invisible to the connectedness condition, (5) above. Instead, the positioning constraint is best stated with respect to its place in the sentence as a whole. This well-formedness condition is given in (8).

(8) The auxiliary must appear in either the first or second position.

For example, consider the positioning of the auxiliary word in the PS for the sample sentence, (2), shown in figure 2.11. As mentioned above, the first two words of this sentence are contained in a single phrase. Because auxiliaries do not combine with other words, the PS for the phrase consists of two structures, one for *ngajulu-rlu* and one for *ka-rna-rla.* As auxiliaries are exempt from the connectedness condition, this phrase is not considered ill-formed. The positioning is checked, however, by the auxiliary positioning condition, which this PS passes, as the auxiliary word is the second structure.

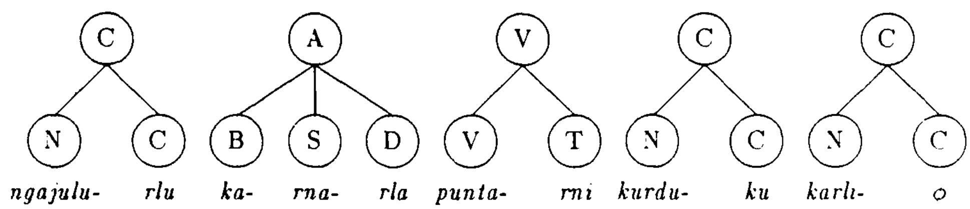


Figure 2.11: The PS for (2).

Some auxiliaries must, in fact, combine with other words. The exceptions consist of the auxiliaries that begin with a clitic morpheme: they must be enclitic to a preceding word. The auxiliary base, *-lpa*, as well as the agreement markers and

the dative registration marker, are all clitics. Auxiliary cliticization differs from normal cliticization (*e.g.*, as with case-markers) in two respects. First, auxiliaries do not have any categorial restrictions on their combinee. Second, as with non-clitic auxiliaries, no combination of structure takes place because clitic auxiliaries are also invisible to the connectedness condition.

A grammatical example of the use of a clitic auxiliary is given in (9). This sentence is just like the sample sentence, except that it is lacking the imperfective base, *ka*. Because the auxiliary word begins with an agreement marker, it must be enclitic to the preceding word, which it is. The PS for this sentence is shown in figure 2.12.

(9) *Ngajulu-rlu-rna-rla punta-rni kurdu-ku karli.*
I-ERG-1s-3d take-NONPAST child-DAT boomerang
'I will take the boomerang from the child.'

Before finishing the discussion of auxiliary positioning, a couple of points should be mentioned. First, note that the requirement the clitic auxiliaries must appear in the second position follows from the requirement that these words be enclitic to a word, and that auxiliaries must appear in either the first or second position (following the well-formedness condition, above). The positioning of short auxiliaries (*i.e.*, those consisting of a single syllable) is also accounted for here. By the condition stated above, such auxiliaries must be enclitic, and so their positioning is similarly handled.

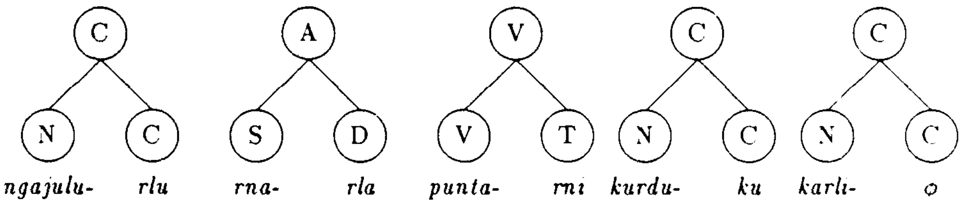


Figure 2.12: The PS for (9).

2.2 Syntax

This section describes the syntactic component of GB. There are three levels of representation within this component, L-structure, D-structure, and S-structure. Both L-structure and D-structure represent the syntactic manifestation of predicative relations. L-structure is concerned with the predicative nature of individual lexical items, while D-structure contains the relations of the sentence as a whole. In fact, it constructed as a combination of constituent L-structures. S-structure, on the other hand, is concerned in part with case-marking relations between case-markers and their arguments. But these two syntactic views are not orthogonal: there is

a tightly constrained relationship between them, given by the mapping from one representational level to another.

An important idea behind these linguistic structures is the notion of *licensing*. The structures contain different elements, some of which allow for the existence of others by licensing them. A prime example of licensing concerns the predicator and its arguments. The presence of an argument in a sentence is due solely to the predicator. At D-structure this licensing concerns the assignment of semantic roles to arguments; only those roles that are part of the predicator's meaning are licensed in the structure. At S-structure, the licensing is for case-marking. As with D-structure, only those arguments selected by the predicator are licensed for case.

Consider the sample sentence, (2). The predicator is the verb stem, *punta*, which, due to its meaning, licenses three arguments, the taker, the taken, and the source from which the taken is taken. *Punta* licenses three positions in D-structure for its arguments. In S-structure, the verb licenses three case phrases (in this instance, marked for ergative, absolutive, and dative case). These licensed arguments appear in the sentence as three case-marked nouns.

All three syntactic levels have the same basic form and contain the same basic entities, *i.e.*, syntactic categories. Their structure is given by $\overline{\text{X}}$-theory, described in the following section. Following this discussion comes a description of each of the sublevels and the mappings between them.

2.2.1 $\overline{\text{X}}$-theory

$\overline{\text{X}}$-theory gives the structure of the syntactic representations of GB. The main idea behind this theory is that each basic item (*e.g.*, noun or verb) is the central element of its own phrase, and that the structure of a sentence consists of a combination of these structures. The central elements are called *heads*, and the structures of which they form the core are called *projections*. The head *projects* some number of levels to form the projection. The highest level of the projection is called the *maximal* projection.

A major claim of $\overline{\text{X}}$-theory is that the same structure schema applies to all categories; all phrases (*e.g.*, noun phrases and verb phrases) are assumed to have roughly the same structure. The number of levels in the projection is parameterized on a per category basis, however. Lexical items, such as nouns and verbs, project two levels;[8] other items, such as case-markers, project one. The auxiliary projects two levels, as explained below. For example, the $\overline{\text{X}}$-structure for verb phrases is depicted in figure 2.13. As with PS, the syntactic structures of $\overline{\text{X}}$-theory are depicted with nodes connected by links; the level of projection is indicated with the number of bars above the categorial label.

The purpose of the projections is to create slots in the structures for the attachment of other projections. These slots are manifest as siblings of the non-maximal projections. The siblings of the head are called *complements*, and the sibling of the first-level projection (for two-level projections) is called the *specifier*. These siblings

[8] In the implementation, nouns do not project any levels; they are left as zero-level nodes. This was done because the parser does not yet cover the predicative use of nominal expressions.

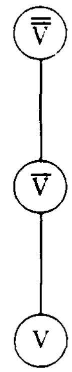


Figure 2.13: The projection for a verb phrase.

in general are called *arguments.* $\overline{\text{X}}$-theory further states that the arguments of a projection must themselves be maximal projections. This is diagrammed in figure 2.14 with a two-level projection.

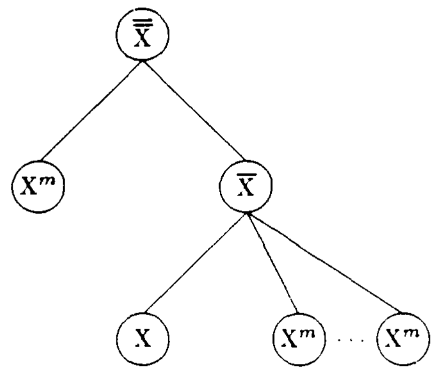


Figure 2.14: A two-level projection and its arguments.

But there is a problem with the traditional theory. $\overline{\text{X}}$-theory states that $\overline{\text{X}}$-structures are ordered by precedence, in addition to hierarchy. How can we account for the free word ordering if the syntactic structures are ordered by precedence? Some perfectly grammatical sentences would be assigned ill-formed structures, having crossing arcs (*e.g.*, when a complement precedes the specifier). The version of $\overline{\text{X}}$-theory presented here is not ordered by sibling precedence. Instead, only the dominance relations are represented; what ordering there is among surface constituents is represented in PS. In line with Occam's razor, syntactic structure need not represent precedence because precedence must be contained in PS to account for the linearity inherent in its level of representation. Though the depictions of syntactic structures must be shown flattened on a page with a direction among the constituent

nodes, it should be remembered that no ordering is implied (*i.e.*, subtrees could be on either side).

Before entering a discussion of the syntactic levels themselves, this section presents two theories that apply to $\overline{X}$-structures. First, the central structural relation of $\overline{X}$-theory, government, is defined. The section concludes with the structural definition of the grammatical functions, subject and object.

Government Theory

The *government* relation has been found to be useful in explaining many syntactic phenomena, such as θ-assignment and case-assignment (explained below). Government is based on the more basic relation of *c-command* (taken from [vRW86]; *cf.*, [Cho81]):

> *C-command:* A c-commands B if and only if the first branching node dominating A also dominates B; and A does not itself dominate B. [p. 142]

For examples of c-command, consider figure 2.14 again. The head c-commands each of its complements, and, in fact, the complements c-command the head. The first-level projection c-commands the specifier, and *vice versa.*

C-command, in turn, is used to define government (also taken from [vRW86]):

> *Government:* X governs Y if and only if Y is contained in the maximal $\overline{X}$-projection of X, X^{max}; X^{max} is the smallest maximal projection containing Y; and X c-commands Y. [p. 291]

Only non-maximal projections may act as governors ('X' in the definition above).[9] Again referring to figure 2.14, we see that only the specifier and complements are governed, and that their sole governors are the first-level projection and the head, respectively.

Grammatical Functions

Grammatical functions are defined in terms of their $\overline{X}$ positions:[10]

> *Subject:* the sibling of the one-level projection, $\overline{X}$, *i.e.*, the specifier
>
> *Object:* a sibling of the zero-level projection, X, *i.e.*, a complement

While there may be any number of objects, as dictated by other aspects of grammar, there may only be a single subject. In fact, predicators are required to have a subject, as assumed by Extended Projection Principle (taken from [Cho86]):

[9]This differs slightly from a more standard notion of *proper* governor, which may only be heads. The difference arises from the particular analysis of Warlpiri, where first-level projections must act as governors.

[10]From these definitions it may seem grammatical functions are unnecessary as they coincide with $\overline{X}$-theory argument positions. However, the definition of subject for languages like English does differ: see [Wil84], for example. To maintain generality the notion of grammatical function is kept distinct. Of course, the dichotomy between languages needs to be worked out

Extended Projection Principle: ... the requirement that clauses have subjects ... [p. 116]

The notions of subject and object are motivated by control phenomena. Control theory attempts to explain the interpretation of the implicit argument that is present in infinitival subordinate clauses. This phenomenon is best introduced by way of an example. (10) below (from [Hal83]) shows two examples of subject control. In instances of subject control the implicit subject of the subordinate clause is understood to be coreferent with the subject of the matrix clause. The presence of the subject-control complementizer, *karra*, attached to the embedded verb, indicates that these sentences are instances of subject control. It is appropriate to refer to the notion of grammatical function, rather than case, because both subject cases (ergative and absolutive) appear in this construction.

(10) a. *Ngarrka-ngku ka purlapa yunpa-rni karli jarnti-rninja-karra-rlu.*
man-ERG IMPERF corroboree sing-NONPAST boomerang
trim-INF-COMP-ERG
'The man is singing a corroboree song while trimming the boomerang.'

b. *Karnta ka-ju wangka-mi yarla karla-nja-karra.*
woman IMPERF-1o speak-NONPAST yam dig-INF-COMP
'The woman is speaking to me while digging yams.'

Parallelling the subject control examples are examples of object control (also from [Hal83]), given in (11). Object control is the phenomenon where the object of the matrix clause, not the subject, is understood to be coreferent with the subject of the subordinate clause. The object-control complementizer, also attached to the embedded verb, is *kurra*. Again, it is proper to employ grammatical functions instead of case because both objective cases (absolutive and dative) are used in this type of sentence.

(11) a. *Purda-nya-nyi ka-rna-ngku wangka-nja-kurra.*
aural-perceive-NONPAST IMPERF-1s-2o speak-INF-COMP
'I hear you speaking.'

b. *Ngarrka-patu ka-rna-jana nya-nyi wawirri panti-rninja-kurra.*
man-PAUCAL IMPERF-1s-333o see-NONPAST kangaroo spear-INF-COMP
'I see the several men spearing the kangaroo.'

c. *Marlu-ku ka-rna-rla wurruka-nyi marna nga-rninja-kurra-ku.*
kangaroo-DAT IMPERF-1s-3d stalk-NONPAST grass eat-INF-COMP-DAT
'I am sneaking up on the kangaroo (while it is) eating grass.'

The control relation is assumed to be a structural one, and therefore the subject/object asymmetry demonstrated in the examples above must be represented in a structural manner. Briefly, this is achieved by placing the subject argument at a higher level in the projection than the object: the subject is associated with the specifier position, and the objects are associated with the complement positions. The choice of subordinate complementizer, either *karra* or *kurra*, then dictates where the

subordinate phrase-marker should be attached to the matrix phrase-marker. In the event of subject control, the subordinate phrase is attached so that it c-commands the subject; for object control, the subordinate phrase c-commands the object.

2.2.2 L-structure

L-structures represent the syntactic manifestation of lexical items. Each item projects into a single L-structure, which is a single $\overline{\mathrm{X}}$-projection. As mentioned above, the number of levels of projection is determined by the item's category. However, there is more that determines the particular manifestation of different lexical items. GB claims that part of the semantic content of a lexical item is involved in the derivation of its syntactic manifestation. In particular, the number and type of its arguments dictates how they will appear. The first part of this section discusses θ-theory, which attempts to explain the mapping from semantic to syntactic arguments. The other part describes the simple set of rules that indicates where the syntactic arguments are placed in the L-structure.

θ-theory

The meaning of lexical items must contain as a minimum information about the number and type of arguments that it takes. Consider the verb from the sample sentence, *punta*. Part of what one knows about taking is that there is a taker, a thing which is taken, and a source from which the taken is taken. Of course, there is more meaning, but this much seems minimally necessary.

θ-theory[11] is concerned with capturing the nature of these semantic arguments and how they appear in syntax. Arguments are called *θ-roles* (thematic roles). Although the theory of θ-roles is still quite fuzzy, a few roles seem to crop up repeatedly. The most common of these are AGENT, the performer of an action, THEME, the object affected by an action, and PATH, the source or goal of an action.

A predicator is said to *select* a number of θ-roles. The list of θ-roles that a predicator selects is called a *θ-grid*. For example, the predicator, *punta*, has a θ-grid that contains three θ-roles, AGENT, THEME, and PATH. However, not all combinations of θ-roles occur in Warlpiri. The θ-grids that do appear are listed in table 2.2.

θ-roles appear as syntactic categories that are said to bear the corresponding role. The theory used here assumes that all arguments appear as *case phrases*.[12] Case phrases receive their θ-role under the syntactic relation of *θ-assignment*, under the relation of government. The position in which they receive their θ-role is called a *θ-position*. An example of θ-assignment will be given in the section below. The list of Warlpiri θ-assigners covered by the parser is given in table 2.3.

Consider the verb *punta* once again. Following table 2.3, we see that both the AGENT and THEME θ-roles are assigned by the verb. The verb is not able to assign

[11] See, for example, [Sto81].

[12] This theory is impoverished, as it does not account for arguments of verbs of belief, for instance. Their arguments can appear as sentential entities, rather than case phrases. Their analysis must be deferred for now.

	θ-grid			example
1.	THEME			*ya* 'to go'
2.	THEME	PATH		*yulka* 'to love'
3.	AGENT	PATH		*warri* 'to seek'
4.	AGENT	THEME		*nya* 'to see'
5.	AGENT	THEME	PATH	*punta* 'to take'

Table 2.2: The five verbal θ-grids.

assigner	θ-role
V	AGENT
V	THEME
DAT	PATH

Table 2.3: Warlpiri θ-assigners.

all of the θ-roles that it licenses, however. The outstanding θ-role, PATH, is assigned by the dative case-marker, DAT. In order to associate this argument with the verbal projection, the verb must *indirectly* assign the θ-role through the dative case-marker. Indirect θ-assignment also takes place under government. This will be demonstrated in the section on D-structure, below.

Placing the Arguments

θ-theory dictates how θ-roles appear syntactically, and which elements license them. The remaining question to answer is where these arguments appear in the L-structure. This information comes from the mapping of θ-roles to grammatical functions. That is, the mapping specifies which argument appears as the subject, and which arguments appear as objects.

The mapping to grammatical function is mediated by the distinction between *external* and *internal* θ-roles[Wil81]. In the standard form of the theory, the external θ-role is assigned its role outside the maximal projection of the predicator, and internal θ-roles are assigned their roles within the maximal projection. Unfortunately, there is no solid theory explaining which θ-roles are external and which are internal. For Warlpiri there is a simple rule that dictates which θ-role of a θ-grid will be external:

> *External θ-role:* If the AGENT θ-role is selected, then it is the external θ-role, otherwise the THEME is.

The theory of external and internal θ-roles differs slightly for the analysis of Warlpiri[Hal83]. The external θ-role, rather than being assigned outside the predi-

cator's projection, is taken to appear in the subject position, and therefore it is also assigned internally. Internal θ-roles are assigned within the projection, and in fact appear as objects.

There is one question that arises: why talk about the subject/object distinction when there is a one-to-one mapping with the external/internal distinction required by θ-theory? That is, the notions of subject and object seem to be redundant. As Williams[Wil84] has pointed out, the external θ-role does not always map to the specifier position in the verb phrase (Warlpiri's subject position). In English, for example, the subject noun phrase is analyzed as the specifier of the projection of INFL. While in Warlpiri the subject does, indeed, appear as the specifier of the verb phrase, we need the distinction between grammatical function and external/internal θ-role in order to maintain cross-linguistic generalization.

As an example L-structure, consider the verb stem, *punta*. Its L-structure is shown in figure 2.15. Following the rule above, we note that its agent θ-role will appear as the subject, and that the others will appear as objects. As shown in table 2.3, the verb stem itself assigns both the agent and theme θ-roles, hence their appearance in the verbal projection. The level at which they appear is dictated by their grammatical function.

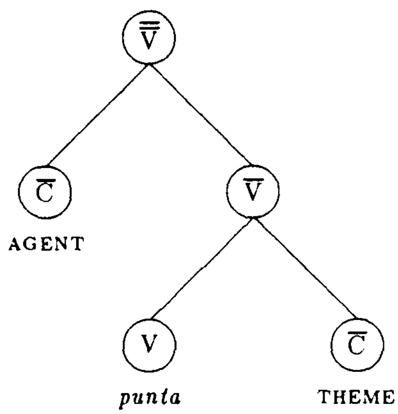


Figure 2.15: The L-structure for *punta*.

Missing from the L-structure for *punta* is the path θ-role. This argument is assigned by the dative case-marker within its own projection, as shown in figure 2.16. Note that while the *appearance* in syntax of this role is licensed by the predicator, the *assignment* of its role is performed by a different element. The two structures are indeed linked together in syntax, as one would expect; this is discussed below in the section on D-structure.

There is a well-formedness constraint in the mapping from θ-grids to L-structure, namely, the θ-Criterion (taken from [vRW86]; *cf.*, [Cho81]):

> *θ-Criterion:* Every chain [*i.e.*, θ-position—MBK] must receive one and only one θ-role. [p. 245]

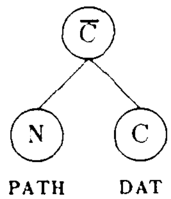


Figure 2.16: The L-structure for DAT.

This principle guarantees that every θ-position, as determined by the lexical information associated with the predicator, will be filled with a θ-role. It furthermore guarantees that each such position will not be filled by more than one θ-role. Observe that this principle has been obeyed in the L-structures for *punta* and DAT, above.

2.2.3 D-structure

D-structure is used to represent the predicate-argument relations of a sentence. D-structures are formed by combining the constituent L-structures of the predicators and arguments. θ-assignment and indirect θ-assignment license combination. Note that these licensing relations are themselves licensed by the semantic content of the predicator. That is, the syntactic relation for assigning a given θ-role may not be present in syntax unless the predicator selects that θ-role.

For an example D-structure, consider the sample sentence, (2), once again. First we examine the core of the sentence, the L-structure for the verb stem, *punta*, given above in figure 2.15. The verb selects three θ-roles, two of which are licensed in the L-structure by means of the θ-assigning functions of the head and first-level projection. The third is licensed by the dative case-marker in its L-structure, as shown in figure 2.16, above. The dative L-structure is licensed, in turn, by the indirect θ-assigning function of the verbal head.

Figure 2.17 depicts the D-structure for the sample sentence. We see that the agent, *ngajulu-rlu*, is attached as sibling of the first level projection; that the theme, *karli*, is attached as sibling of the verbal head; and that the path, *kurdu-ku*, is attached as sibling of the dative case-marker, which itself is attached as a sibling of the verbal head. Every L-structure has been properly licensed because each has been incorporated into the structure.

2.2.4 S-structure

S-structure represents a different syntactic view than D-structure, essentially that of case-marking. Case-marking associates predicates with their arguments at the level of S-structure. This licensing relation is parallel to the relation of θ-assignment.[13]

[13] For some languages, such as English, the notion of abstract case has been proposed to account for case-marking phenomena that do involve an overt case-marker. Abstract case is assumed to account for nominative and accusative case-marking, where the verb assigns these cases to the

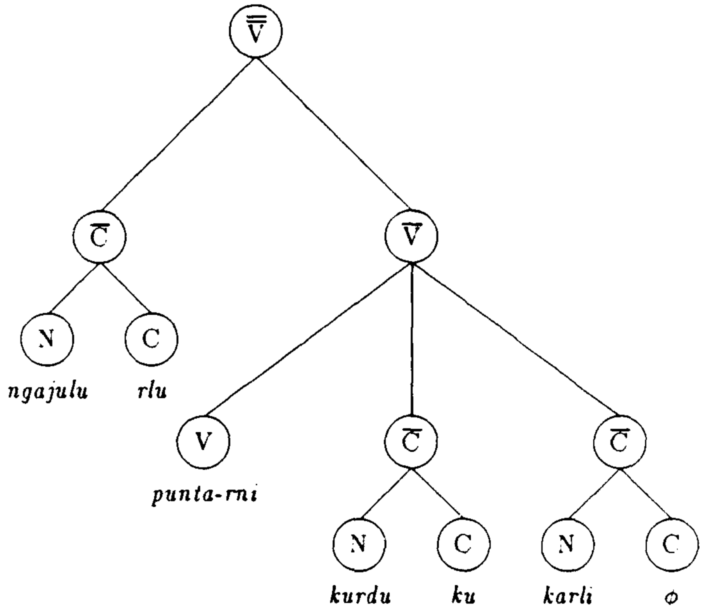


Figure 2.17: The D-structure for sentence (2).

This section begins with a discussion of Case theory. After this is a section on the placement of case-marking relations in the structure. Lastly, the section discusses how to represent the auxiliary in S-structure.

Case Theory

At S-structure, arguments are associated with their predicates via case. As with semantic selection of θ-roles, predicators determine their case *subcategorization*, *i.e.*, the cases that they license.[14] The set of cases for which a predicator subcategorizes is called a *case array*. For example, the case array for *punta* contains all three syntactic cases, ergative, absolutive, and dative. In Warlpiri there are five groups of verbs that have different case arrays, shown in table 2.4. These five classes are derivative from the list of possible θ-grids found in Warlpiri.

The S-structure association of arguments with predicators is effected in a two-part relation. Arguments—which, for this thesis, are only nouns—are associated with a case-marker by the relation of case-marking. Case phrases, in turn, are associated with argument positions in the predicator's projection by the relation of case-assignment. Both of these relations obtain under the relation of government, as with θ-assignment in D-structure.

subject and object, respectively. The parser should be able to incorporate abstract case by giving case-marking capabilities to lexical items other than just case-markers.

[14]In fact, the cases for which a predicator subcategorizes are derivable from the θ-roles they select. The mapping is presented in the following section.

	case array			example
1.	ABS			*ya* 'to go'
2.	ABS	DAT		*yulka* 'to love'
3.	ERG	DAT		*warri* 'to seek'
4.	ERG	ABS		*nya* 'to see'
5.	ERG	ABS	DAT	*punta* 'to take'

Table 2.4: The five verbal case arrays.

In Warlpiri there are three syntactic case-markers (ERG, ABS, and DAT), corresponding to each of the syntactic cases: ergative, absolutive, and dative. Naturally enough, each syntactic case-marker marks its argument noun phrase for its own case. The case-markers and their phonetic realizations are presented in table 2.5. The list of Warlpiri case-assigners is given in table 2.6. (Note that 'T' stands for 'tense element'.)

marker	case	phonetic realizations
ERG	ergative	*-ngku*, *-ngki*, *-rlu*, *-rli*
ABS	absolutive	ϕ
DAT	dative	*-ki*, *-ku*

Table 2.5: The Warlpiri case-markers.

assigner	case
V	ergative
T	absolutive
DAT	dative

Table 2.6: The Warlpiri case-assigners.

Figure 2.18 shows an example of the case-marking of an ergatively marked noun phrase. *Rlu* is the case-marker and *ngajulu* is being marked for case. Note that *rlu* does, indeed, govern its argument, *ngajulu*.

Placing the Arguments

Case theory dictates how cases are licensed in S-structure and how arguments are manifest (*i.e.*, through the relations of case-marking and case-assignment), but it

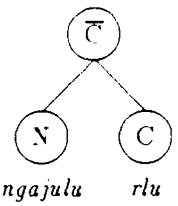


Figure 2.18: An example of ergative case-marking.

does not indicate where they are to be placed. Specifically, it does not indicate the grammatical functions corresponding to the cases that have been licensed. There is a simple set of rules that determines the mapping between case and grammatical function[Hal83]:

1. Identify the subject function with the ERG argument, if there is one, otherwise with the ABS argument.
2. Identify the object function with the DAT argument, if there is one, otherwise with the ABS argument (if this is not already identified as the subject).

Figure 2.19 shows the S-structure for (2).[15] Observe that each of the noun phrases has been marked for its case by the appropriate case-marker, by virtue of their governed status. Observe further that the case phrases have been assigned their case by the appropriate case-assigners. For the ergative and absolutive arguments, the assignment was performed by the first-level projection of the verbal head and the tense element, respectively; for the dative argument, the assignment was performed by the dative case-assigner.

Auxiliary Syntax

The syntax of the auxiliary is somewhat *ad hoc*, due to its ill-understood nature. Its main function is to combine with the rest of the sentence in two ways: it combines with the tense of the verb to add aspect information; and, it combines with the arguments for the purpose of person and number agreement. The structure of the auxiliary facilitates the combination (which is discussed in the section on semantic interpretation, below). The base is considered to be the head of the auxiliary. The nominal agreement clitics are taken to be its objects, and the verbal projection is taken to be its subject. As an example, the complete S-structure for the sample sentence is depicted in figure 2.20.

2.2.5 The Mapping Between S-structure and D-structure

The levels of D-structure and S-structure represents the syntax of a sentence from two different aspects. D-structure is concerned with assignment of θ-roles; S-

[15] The auxiliary projection is not shown here, as it will be discussed below.

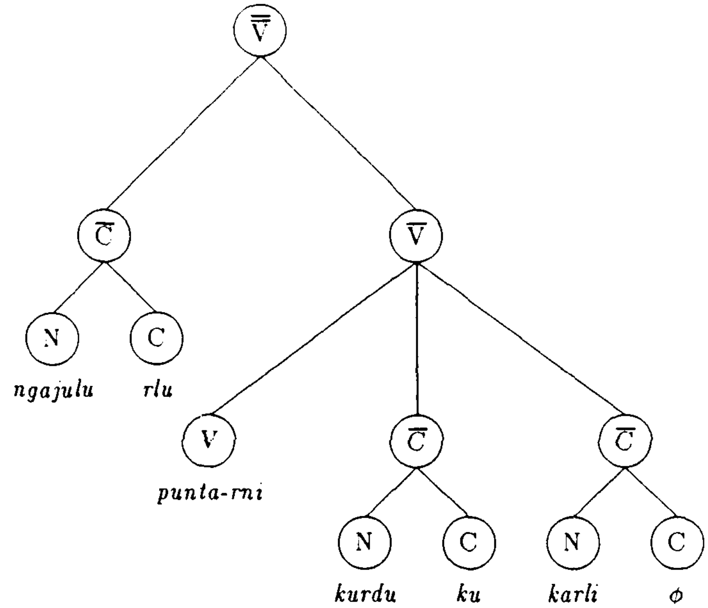


Figure 2.19: Most of the S-structure for sentence (2).

structure represents case-marking relations. Both of these levels, however, are concerned with the relation between a predicator and its arguments as manifested in syntax. GB posits that the two levels are indeed related, and that one structure can be transformed into the other by movement. That is, nodes that exist in one place in D-structure in order to receive their θ-roles may move in the structure in order to receive the corresponding case.[16] This licensing requirement for arguments is given by the Case Filter[Cho81]:

> *Case Filter:* *NP[17] if NP has phonetic content and has no Case [p. 49]

Movement is allowed by the very simple rule of Move-α, stated in (12) (taken from [vRW86]; *cf.*, [Cho81]).

(12) Move any category α anywhere.

This rule must be restricted, however, so it won't massively overgenerate ungrammatical sentences. One strong constraint is the Structure-Preserving Hypothesis, which in part limits the range of grammatical transformations (taken from [Emo76]):

[16]There are other reasons for movement, such as *wh*-movement. These phenomena lie outside the purview of the parser, so they won't be covered here.

[17]'NP' is the traditional notation for a noun phrase. It corresponds to the maximal projection of N, $\overline{\overline{N}}$.

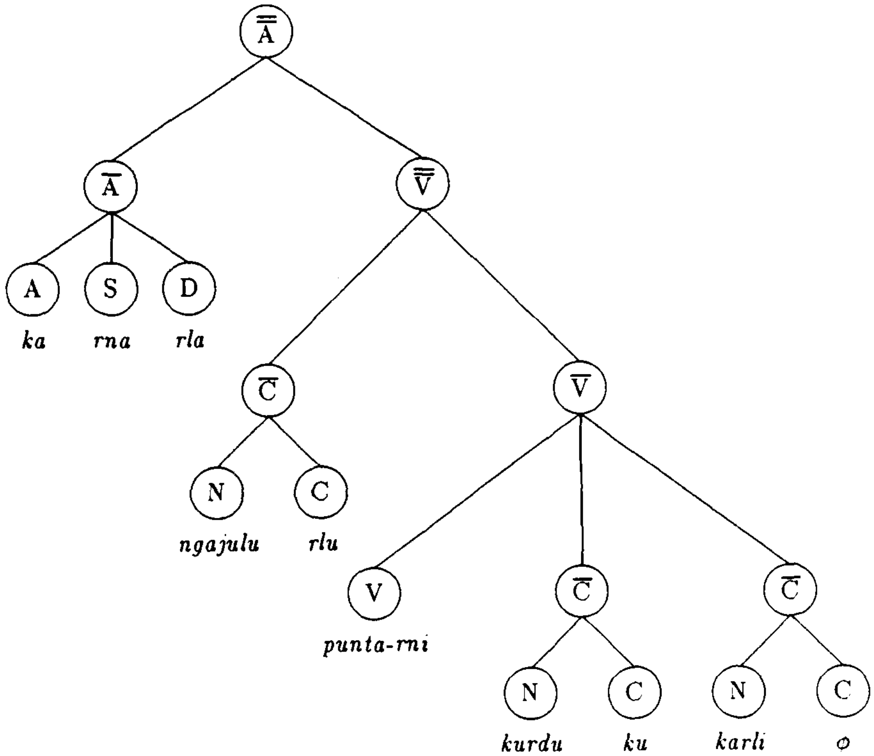


Figure 2.20: The S-structure for sentence (2).

... a transformational operation is structure-preserving if it moves, copies, or inserts a node C into some position where C can be otherwise generated by the grammar. [p. 3]

For the simple phenomena handled by the parser, this really boils down to the movement of case phrases. Following the constraint, the parser should only allow movement of case phrases into S-structure positions that are licensed at D-structure. Roughly speaking, arguments may only move into argument positions.

Another limit to movement is the Projection Principle, which constrains the possible mappings between the argument positions of S-structure, D-structure, and LF (borrowed from [vRW86]; *cf.*, [Cho81]):

Projection Principle: The θ-Criterion holds at D-structure, S-structure, and LF. [p. 252]

This principle ensures consistency between these three levels of representation (for the parser, just the levels of D-structure and S-structure). It establishes a connection for each of the predicator's argument positions at each of the levels. Given a case phrase that has been assigned a θ-role in a certain position in D-structure, it would be inconsistent for another case phrase to move into that position, in a sense usurping the θ-role assigned there.

In Warlpiri, however, Move-α is rarely used. It seems that the parameterization of the language is constrained so that Move-α need apply only in a few, select instances. As for the parser, the simple sentences in its domain do not call for Move-α at all because arguments need not move to receive case: the positions where they are assigned their θ-roles are the same positions where they are assigned the corresponding case. This can be seen for the mapping from θ-roles to cases, shown in table 2.7.[18]

θ-role	case
AGENT	ERG
THEME	ABS
PATH	DAT

Table 2.7: The θ-role/case mapping for Warlpiri.

Thus movement need not enter into the theory on which the parser is based. Because of this, the S-structures and D-structures look the same, except for the syntax of the auxiliary, which is represented only in S-structure.

[18] Unfortunately, there is no explanatory theory for this mapping. The descriptive theory of this mapping is simply due to empirical studies.

2.3 The Lexicon

The lexicon maps surface string entities into their lexical entries containing information dictating their manifestations in both precedence and syntactic structure. A complete entry contains information that is both specific to the lexical item, as well as applicable to the lexical classes of which it is a member. The lexicon contains two structures for representing each kind of lexical information. The first is a mapping from items to entries containing specific information, and the second is a set of rules that applies to classes of lexical items.

Lexical entries contain category, precedence, syntactic, and semantic information. Categorial information refers to the lexical item's part of speech. Included in the scope of the parser are nouns, number-markers, syntactic case-markers, verbs, tense elements, and auxiliary components (bases, subject and object clitics, and dative registration markers).

The precedence and syntactic components of lexical entries dictate their participation in the corresponding structures. These structures also refer to the categorial information of the entry. For example, case-markers may only combine with nouns; that is, the category of the combinee is relevant to PS. In SS, the category of the item determines the number of levels of projection.

The last component, semantic information, manifests itself both in syntax and in semantic interpretation. Semantics is connected to syntax through the mapping of θ-roles to syntactic categories, as well as cases (described above). The person and number of nouns, pronouns, and auxiliary agreement clitics are prime examples of *interpretive information stored in the lexicon.*

2.3.1 The Lexical Entry

Lexical entries contain information for each of the four components listed above. A summary of the individually stored information is given in table 2.8. Note that categorial information is always contained in the individual entry because it is particular to the lexical item.

The precedence information stored on a per item basis is rather small. Verbs and tense elements contain their conjugation class (one to five), which is used during inflected verb analysis. Auxiliary bases contain their number of syllables, used for checking the well-formedness of words.

Syntactic information concerns both S-structure and D-structure. The case marked by a case-marker depends on the particular case-marker and so it is stored in the entry. The case and θ-role assigned by the dative case-marker, unlike other predicators, is also stored in the entry.

The semantic component is also straightforward. The person and number information for pronouns and agreement clitics depends on the particular lexical item, so it is stored individually. The same is true for the tense information of tense elements, and the aspect of auxiliary bases.

precedence	conjugation class (verbs and tense elements)
	syllables (auxiliary bases)
syntactic	case-marking (case-markers)
	case-assignment (dative case-marker)
	θ-assignment (dative case-marker)
semantic	person (pronouns and agreement clitics)
	number (pronouns, number-markers, agreement clitics)
	tense (tense elements)
	aspect and tense restrictions (auxiliary bases)
	θ-grid (predicators)

Table 2.8: The elements of an individual lexical entry.

2.3.2 Lexical Rules

Lexical rules encode descriptive information that applies to classes of lexical items. Each rule is in a simple "if–then" form. The conditional part tests for membership in a given class, and the action part indicates the information to be added to the cumulative lexical entry if the membership is satisfied. The classes of information represented with lexical rules are listed in table 2.9.

precedence	adjacency requirements
	directed argument identification
syntactic	case-assignment (predicators)
	θ-assignment (predicators)
	θ-linking (predicators)
	projection (predicators)

Table 2.9: The range of lexical rules.

The adjacency requirements for the various categories are listed below. For most categories the requirements follow straightforwardly from the data. The auxiliary relies on the more involved notion of a linear template, as described above in the section on PS.

- Number-markers must be enclitic to nouns.
- Case-markers must be enclitic to nouns.
- Tense elements must be enclitic to verbs.

- Auxiliary words may be enclitic to any word.
- Auxiliary components must be enclitic to each other according to the template given in figure 2.9.

The only instance of directed argument identification handled by the parser occurs with case-markers. Case-markers take both the nouns to which they are enclitic and the preceding nouns within their phrase as arguments. This is stated in the rules below.

- Case-markers take the nouns to which they are enclitic as arguments.
- Case-markers take preceding nouns as arguments.

The rules for case-assignment and θ-assignment were given in the form of tables above. They are encoded here in the lexicon as rules. The other component of the syntactic information dictates the levels of projection for each category, listed below. Verbs project two levels in order to create slots for subjects and objects. Case-markers project one level as their only arguments are the nouns that they mark. As mentioned above, auxiliaries project two levels, one for the agreement clitics, and one for the argument verb phrase.

- Verbs project two levels.
- Case-markers project one level.
- Auxiliary bases project two levels.

2.4 Semantic Interpretation

The semantic interpretation performed by the parser operates on both S- and D-structure, depending on the type of interpretation involved. For example, argument identification is read off of D-structure, using the θ-assignment relation. Agreement with the auxiliary is checked at S-structure, as the auxiliary is not represented at D-structure. This section discusses the semantic interpretation within the domain of the parser: argument identification, interpretation of null elements (null auxiliary components and null anaphora), and other semantic well-formedness conditions.

2.4.1 Argument Identification

Argument identification is the association of surface string components with the semantic functions that they fill. As this interpretation concerns the semantic aspect of the syntactic structure, D-structure is used here. Arguments are identified by virtue of having been assigned a θ-role licensed by the predicator. Since this information is represented explicitly, the interpretation of the syntactic structure is straightforward. For example, argument interpretation for the sample sentence gives us the results in table 2.10.

θ-role	word
AGENT	*ngajulu* ('I')
THEME	*karli* ('boomerang')
PATH	*kurdu* ('child')

Table 2.10: The θ-role/word mapping for (2).

2.4.2 Null Auxiliary Components

As mentioned in the introduction, any part of the auxiliary word may be phonologically null. This does not mean that the corresponding information is missing, however. Each auxiliary component has a default value that is applied in the absence of overt morphemes to the contrary. The null auxiliary base, following the table presented in the introductory chapter, indicates perfective aspect. Null auxiliary agreement clitics have default values of third-person, singular. It should be noted that their interpretation depends on the verb's subcategorization frame. If, for example, the verb does not subcategorize for an object, then there is no default interpretation for object agreement.

2.4.3 Null Anaphora

In Warlpiri none of the arguments of a verb need be expressed by case phrases: no argument need be associated with an argument position in S-structure. When a case phrase is absent from the surface string, the corresponding registration clitic in the auxiliary takes on more of a pronominal character. Such clitics would be translated as 'I' or 'her,' for example. (13) gives an example sentence in which none of the arguments are overtly expressed as a case phrase. In some contexts, this would be the preferred mode of expression, and inserting overt pronouns would give an emphatic reading.

(13) *Punta-rni-rna-rla.*
take-NONPAST-1s-3d
'I may take him/her/it from him/her/it.'

2.4.4 Semantic Well-formedness

There are three conditions on semantic well-formedness concerning the licensing of and agreement with components of the auxiliary. Note that because the conditions involve the auxiliary, this interpretation is performed on S-structure. The first condition, given in (14), is rather straightforward.

(14) The auxiliary base must be compatible with the tense of the inflected verb.

As an example of a grammatical use of the auxiliary, consider the sample sentence once again. Table 2.11 shows the tense correspondence between the auxiliary and

the inflected verb. Because the tense of the verb meets the tense restrictions of the auxiliary, the sentence is considered well-formed from this point of view.

base	compatible tenses	tense element	tense
ka	non-past	*rni*	non-past

Table 2.11: The tense correspondence for (2).

The second condition concerns the licensing of the agreement clitics. As with arguments, the semantic argument selection of the predicator dictates which clitics are licensed. This is stated in (15). Note that agreement clitics are best formulated in terms of grammatical function—not case—lending more support to the concept of grammatical function (in addition to control facts as mentioned above). In accordance with the Extended Projection Principle (the requirement that predicators have subjects), there must always be a subject, so this agreement clitic is always licensed. Thus, this condition really serves as a condition on the appearance of the object clitic and the dative registration marker.

(15) Nominal agreement clitics must be licensed by the main predicator.

The last condition, presented in (16), also follows from observed data. For example, consider the agreement correspondence of the sample sentence, shown in table 2.12. This sentence is also well-formed with respect to this condition. The subject clitic, *rna*, is first-person singular, which agrees with the subject pronoun. *ngajulu*. The object clitic is null, and therefore defaults to third-person singular (as described above), which agrees with the object noun, *kurdu*. Because *kurdu* is unmarked for number, it agrees with either singular or plural; the corresponding clitic disambiguates between the two.

(16) The nominal agreement clitics of the auxiliary must agree with the arguments of the main verb in person and number.

GF	clitic	person/number	argument	person/number
subject	*rna*	first-person singular	*ngajulu*	first-person singular
object	ϕ	third-person singular	*kurdu*	third-person sing. or plural

Table 2.12: The agreement correspondence for (2).

Note that the agreement correspondence shows a semantic gap in Warlpiri. Because the auxiliary has only two positions in it for nominal agreement, there is no

agreement with the third argument when selected by the predicator. This makes it impossible to say something like 'I take you from him' without supplying the overt pronoun, *nyuntu* 'you.' (17) shows the translation of this sentence without the overt pronoun; it must be interpreted with a third-person direct object, due to the lack of registration in the auxiliary.

(17) *Punta-rni-rna-ngku-rla.*
take-NONPAST-1s-2d
'I may take him/her/it from you.'

Chapter 3

Representation and Algorithm

This chapter presents a complete description of the representations and algorithms of the parser. The goal of this presentation is to show how the parser handles both free- and fixed-order phenomena. To demonstrate this ability, I will show the parser processing the sample sentence in (1) (repeated here from the introduction) and some of its permuted cousins. Specifically, we will see that the parser derives equivalent syntactic structures, from which equivalent semantic interpretations can be retrieved.

(1) *Ngajulu-rlu ka-rna-rla punta-rni kurdu-ku karli.*
I-ERG IMPERF-1s-3d take-NONPAST child-DAT boomerang
'I am taking the boomerang from the child.'

Ordering phenomena do not constitute the only domain of the parser. however. A more nearly inclusive list of the Warlpiri phenomena that are handled is given in table 3.1. The discussion below will also demonstrate how the parser computes each of these phenomena.

precedence	nominal, verbal, and auxiliary composition continuous case phrases auxiliary positioning
syntax	grammatical functions free phrase order
semantics	argument identification null anaphora null auxiliary components tense and argument agreement

Table 3.1: Phenomena handled by the parser.

When reading the descriptions below it is important to remember which structures are responsible for which phenomena. The phenomena involving precedence

are processed with precedence structure (PS); the syntactic phenomena are handled by syntactic structure (SS). Semantic processing is accomplished with a set of interpretive routines that operate on SS.

The next section discusses the representations of both PS and SS, as well as the lexicon. Section two presents the algorithms, and demonstrates their ability to handle the phenomena listed above. The last section gives a trace of parsing the sample sentence.

3.1 Representation

The parser was designed in an object-oriented style because it seems to capture the nature of Government-Binding based processing. Two major objects in the parser are precedence structure (PS) and syntactic structure (SS). The other major object is the lexicon, which is the repository of information for each lexical item. First I discuss the output structures, and then the lexicon.

3.1.1 Precedence Structure and Syntactic Structure

Both PS and SS are based on trees. Each node in a tree contains a category label, and data and actions particular to the level of representation. For example, in PS there are actions for combining adjacent nodes. In SS, on the other hand, actions may not use precedence information because it is not represented at that level. Instead, there are actions for combination of syntactic structures such as case-marking and θ-assignment.

PS is actually an ordered forest of ordered trees. Each tree represents parts of the input sentence where precedence is relevant, such as among the morphemes of a word. The relation between the trees in the forest is not relevant to processing the sentence; however, the ordering is kept to mimic the order of the input sentence. Because phrases are not ordered with respect to one another, the PS for Warlpiri sentences will not contain trees with two phrases in them; rather, there will be one phrase per tree.

SS, on the other hand, is an unordered forest of unordered trees; only hierarchy is represented here. The need for a forest rather than a single tree is a bit subtle. Following the GB principle of Full Interpretation—the requirement that every element of syntactic structure receive an interpretation—we would expect that grammatical sentences correspond to a single structure in syntax; that is, that no element be left unattached because it isn't licensed. This is, indeed, a condition of grammaticality, and the parser checks this upon completion of the parse. However, a forest is required because during the parse there may be several unconnected trees corresponding to different parts of the input sentence. This is a key to the processing of free order phenomena. Consider (2), which is a variation of the main example sentence; its input representation is given in (3).

(2) *Kurdu-ku ka-rna-rla ngajulu-rlu karli punta-rni.*
child-DAT IMPERF-1-3 I-ERG boomerang take-NONPAST
'From the child I am taking the boomerang.'

(3) (((KURDU KU) (KA RNA RLA)) ((NGAJULU RLU)) ((KARLI))
((PUNTA RNI)))

In the process of parsing this sentence, which is performed left-to-right, the parser will reach a stage where it has processed all but the last word, *punta-rni*. At this point it will have parsed the auxiliary word, as well as each of the three case phrases. Because the verb has not yet entered the parse, there will be no way for the substructures to be connected; instead, they must reside separately, as shown in figure 3.1.[1] When the verb does enter the syntactic structure, the arguments may be connected by inserting them into the argument positions of the verb's projection.

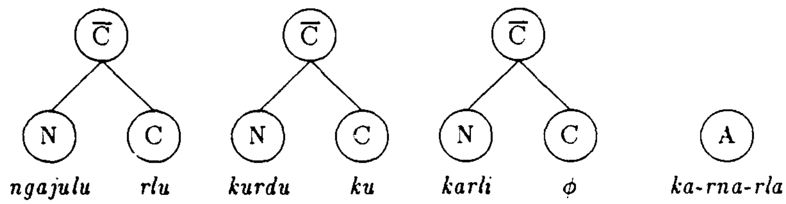


Figure 3.1: The SS after parsing four words of (3).

One other difference between PS and SS concerns *projections*. Each morpheme in the input sentence is projected into PS as a single node. However, in SS syntactically relevant parts of the input sentence project into L-structures, which may contain zero, one or two levels of projection. L-structures are encoded in SS with the aid of the "projection?" flag that is stored with each node. This flag is true if and only if its parent is a member of its projection. This requires, of course, that exactly one of a node's children have a true projection flag. This well-formedness condition is met by the construction of SS, as explained in section two.

An example should clarify the representation. The syntactic manifestation of a case phrase consists of a case-marker that has projected one level, taking the constituent nouns as arguments. Consider the case phrase in (4). The case-marker, *rlu*, has three argument nouns, *yirrinji*, *yirraru*, and *kardirrpa*. This is shown graphically in figure 3.2.

(4) *yirrinji yirraru kardirrpa-rlu*
centipede homesick brave-ERG
'the brave, homesick centipede'

The parser's representation for the same phrase is given in figure 3.3. The parser displays its results on its side, so that the top of the projection appears to the left. Note that the left-most node's category is case, as it is the first-level projection of the case-marker, *rlu*, shown at the bottom of the structure. Note also that the

[1]The syntactic structure for the auxiliary has been glossed as a single node. The details of auxiliary structures are given below.

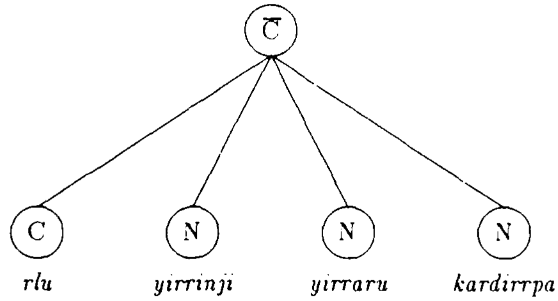


Figure 3.2: The syntactic manifestation of (4).

projection flag for the case-marker is true, indicating that it is the child of left-most node that is in its projection; the other nodes have projection flags that are false.

<u>SS</u>

```
projection?: NIL
category: CASE
children: projection?: NIL
          morpheme: YIRRINJI
          category: NOUN

          projection?: NIL
          morpheme: YIRRARU
          category: NOUN

          projection?: NIL
          morpheme: KARDIRRPA
          category: NOUN

          projection?: T
          SS data: CASE-MARKED: ERGATIVE
          morpheme: RLU
          category: CASE
```

Figure 3.3: The SS for (4).

3.1.2 The Lexicon

The lexicon is represented simply as a set of pairs of lexical items and entries.[2] Each entry contains categorial, precedence, syntactic, and semantic information. Entries

[2] As in other languages, Warlpiri does exhibit some lexical ambiguity. However, the parser does not handle this phenomenon. So, it is assumed that each lexical item maps into exactly one entry. This shortcoming is discussed in the conclusion.

contain only the information that is particular to the lexical item. For example, pronouns contain their person and number information, but nouns do not because they all have default values of third-person, singular.[3]

Figure 3.4 shows the lexicon used by the parser to parse the sample sentence, (1). Each lexical item is given the category under which it is listed. For example, *karli*, *kurdu*, and *ngajulu* are all declared to be nouns. Each item has associated with it optional information, which may be either for PS or SS; the information for SS is both syntactic and semantic in nature. This information may come either in the form of data or actions.

A few of the entries are highlighted here; the remainder of this lexicon will be discussed below in the section on algorithms. The pronoun, *ngajulu*, is distinguished from other nouns because its person and number information is stored in the lexicon. It needn't have a different category, though, because nouns and pronouns act alike in both precedence and syntax structures.[4] Associated with each auxiliary element is the number of syllables it contains. This information is used during word parsing to check well-formedness: each word in Warlpiri must consist of at least two syllables.

Much lexical information applies not to a single item but to entire classes of items. For example, all verbs in Warlpiri that select an agent θ-role assign ergative case. Since this case-assignment is a feature of all such verbs, it wouldn't be appropriate to store the action in each verb's entry; instead, it is stated once, as a rule. These rules are represented straightforwardly as a list of pattern-action rules. After lexical look-up is performed, the list of rules is applied. If the pattern of the rule matches the category, the rule fires, and the information specified in the "action" part of the rule is added to the node.

For example, consider the lexical rules that encode the manifestation of θ-grids in SS. The first set of rules, shown in table 3.2, indicates the number of levels of projection for the L-structures of certain categories.

If	the item is a case-marker then it projects one level.
If	the item is a verb then it projects two levels.

Table 3.2: Lexical rules for projection.

The next set of rules concerns the licensing of case-assignment actions. That is, these elements must be present in the sentence for the case-assignment action that

[3] Actually, as mentioned in the introductory chapter, nouns have a default number of either singular or plural. Combination with the agreement clitics of the auxiliary determines which.

[4] In later versions of the parser, this distinction may well have to be implemented in the categorial information because pronouns can enter into some syntactic constructions that nouns cannot. For example, in English it is perfectly acceptable to have a noun phrase consisting of a pronoun by itself; it is ungrammatical simply to have a noun. Nouns must either be plural or have a determiner appear with them.

```
(noun
  (KARLI)
  (KURDU)
  (NGAJULU
    (ss (data (person 1)
              (number (singular))))))
(case
  (KU
    (ss (data (case-assigned dative)
              (case-marked dative)
              (theta-assigned path))))
  (RLU
    (ss (data (case-marked ergative)))))
(verb
  (PUNTA
    (ps (data (conjugation 2)))
    (ss (data (theta-roles (agent theme path))))))
(tense
  (RNI
    (ps (data (conjugation 2)))
    (ss (data (tense nonpast)))))
(auxiliary-base
  (KA
    (ps (data (syllables 1)))
    (ss (data (tenses (nonpast))
              (aspect imperfect)))))
(auxiliary-subject
  (RNA
    (ps (data (syllables 1)))
    (ss (data (person 1)
              (number singular)))))
(auxiliary-dative
  (RLA
    (ps (data (syllables 1)))))
```

Figure 3.4: The portion of the lexicon needed for parsing sentence (1).

it licenses to become manifest in the structure (*i.e.*, added to the SS actions of the projection). The rules are given in table 3.3.

If the item selects an agent θ-role
then it licenses assignment of ergative case.

If the item is a tense element
then it licenses assignment of absolutive case.

If the item is a dative case-marker
then it assigns dative case.

Table 3.3: Lexical rules for case-assignment.

The third set of rules is for the licensing of θ-assignment, as given in table 3.4. These rules also show the path θ-assigning property of dative case-markers. This θ-role is combined syntactically with the predicator's L-structure via the indirect step of θ-linking which operates on nodes that have been θ-assigned.

If the item selects an agent θ-role
then it licenses assignment of that role.

If the item selects a theme θ-role
then it licenses assignment of that role.

If the item selects a path θ-role
then it licenses linking of that role.

If the item is a dative case-marker
then it assigns the path θ-role.

Table 3.4: Lexical rules for θ-assignment.

The above two tables gave rules largely for licensing case-assignment and θ-assignment actions. The rules in table 3.5 dictate *where* the actions are to be situated in the L-structure; these rules determine grammatical function. When these rules talk about being manifested as a certain grammatical function, it means that the actions for the corresponding case-assignment and θ-assignment are both placed at that level in the projection.

An example should help to clarify the operation of these rules. Consider the parsing of the verb stem, *punta*. In the first step of lexical look-up, the item's entry is retrieved from the lexicon; the entry is shown in figure 3.5.

In the second step lexical rules are applied to the entry. Looking to the first

If an agent θ-role is selected
then it will be manifest as the subject.

If a theme θ-role is selected and there is already a subject
then it will be manifest as an object.

If a theme θ-role is selected and there is no subject
then it will be manifest as the subject.

If a path θ-role is selected
then it will be manifest as an object.

Table 3.5: Lexical rules for determining grammatical function.

```
(PUNTA
  (ps (data (conjugation 2)))
  (ss (data (theta-roles (agent theme path)))))
```

Figure 3.5: The lexical entry for *punta*.

set of rules in table 3.2, we find one rule that applies: verbs project two levels in syntax. In the next set of rules for case there is one rule that applies. Looking to the verb's θ-grid, we see that it does select an agent θ-role; according to the rule, the verb therefore licenses assignment of ergative case.

In the set of rules concerning θ-assignment three rules apply. *Punta* selects an agent, a theme, and a path; therefore, according to these rules, it licenses θ-assignment of all three. More precisely, it licenses assignment of the first two. The path θ-role is actually assigned by the dative case-marker; the verb indirectly assigns the θ-role via θ-linking.

The last set of rules determines the grammatical function of each of the arguments. Following the first rule of the set we see that the agent θ-role will appear in the subject position. This means that case assignment of its case, ergative, and θ-assignment of its role, agent, will take place in the specifier position. The other two θ-roles will appear as objects, so their case- and θ-assignment actions will be placed in the zero-level projection of the verb, as sibling of the complement positions. The computed entry is shown in figure 3.6.[5]

The interaction between licensing, case and θ-role operations is best illustrated by an example. Figure 3.7 shows the morphological parse for *punta*. The PS component of the output is rather straightforward, so I continue with SS. First notice that

[5]This entry may either be computed as the lexical item is entered into the lexicon, or upon look-up of the item during the parse. The parser, in fact, takes the former tack in order to save time during parsing.

```
(PUNTA
  (ps (data (conjugation 2)))
  (ss (data (theta-roles (agent theme path)))
      (actions (projections 2)
               (license (case-assign . ergative))
               (license (theta-assign . agent))
               (license (theta-assign . theme))
               (license (theta-link . path))
               (specifier (case-assign . ergative))
               (complement (case-assign . absolutive))
               (complement (theta-assign . agent))
               (complement (theta-assign . theme))
               (complement (theta-link . path))
```

Figure 3.6: The computed entry for *punta.*

punta has projected two levels, according to the specification in its `projections` action. Once projected, the SS parser places *punta*'s syntactic actions in its structure. The actions `specifier` and `complement` place these assignment actions at the first-level and zero-level nodes in the verb's projection, respectively. In order for `specifier` and `complement` to execute there must be a corresponding `license` action in the structure. *Punta* itself licenses all but one action, namely, the case-assignment action for absolutive case, which is actually licensed by tense elements. As a result this `complement` action must await the arrival of the tense element in order for it to fire and place the case-assigning action in the structure. All of the other placement actions have fired, however.

3.2 Algorithm

An overview of the parser's operation is given in figure 3.8. Input sentences are given to the PS parser that traverses them left-to-right and builds up PS. Every time a unit of PS becomes syntactically relevant—as determined by the item's lexical entry—the unit is sent to the syntactic parser. The syntactic parser accepts the incoming unit, projects it according to its lexical information, and then enters it into SS. When finished, the syntactic parser returns control to the PS parser which consumes some more input. Upon completing the input sentence, the PS parser stops, and both output structures are returned.

First I discuss the PS parser, and demonstrate its operation with examples that cover the range of phenomena involving precedence. Then the syntactic parser is discussed, again with examples showing its ability to handle the advertised syntactic phenomena. This section concludes with a presentation of the semantic interpretation that the parser performs on the output syntactic structure.

PS

```
0: data: CONJUGATION: 2
   morpheme: PUNTA
   category: VERB
```

SS

```
projection?: NIL
category: VERB
children: projection?: T
          actions: CASE-ASSIGN: ERGATIVE
                   THETA-ASSIGN: AGENT
          category: VERB
          children: projection?: T
                    actions: THETA-LINK: PATH
                             THETA-ASSIGN: THEME
                             COMPLEMENT: (CASE-ASSIGN . ABSOLUTIVE)
                    data: THETA-ROLES: (AGENT THEME PATH)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          PATH: DATIVE
                          SUBJECT: AGENT
                          OBJECT: PATH
                    morpheme: PUNTA
                    category: VERB
```

Figure 3.7: The PS and SS for *punta*.

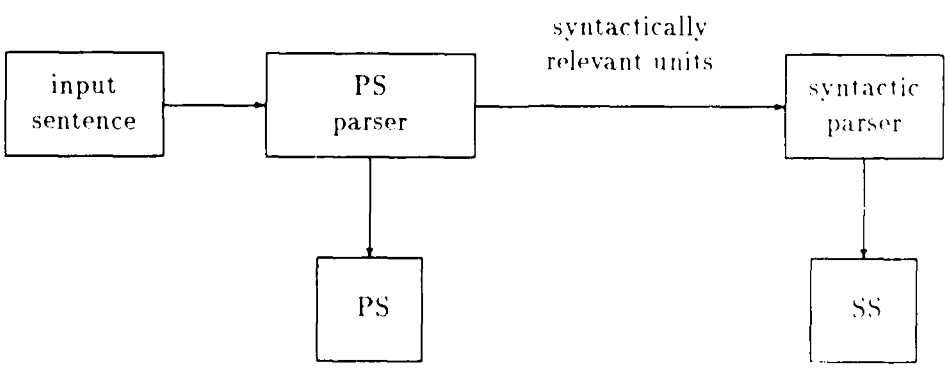


Figure 3.8: An overview of the parser's flow of control.

3.2.1 The PS Parser

In the first part of this section I discuss the algorithms of the PS parser. Following this is a sequence of examples demonstrating its range of coverage.

The Basic Engine

The PS parser is a recursive engine that operates on the four phonological levels of the input sentence. The top level of the parser accepts the entire sentence as input. It calls on the phrasal parser to parse each of the constituent phrases, and then performs sentential actions on the returned phrasal structures. In a like manner, the phrasal parser calls on the word-level parser to parse constituent words. The word-level parser calls on the morphological parser which is essentially the look-up routine for the lexicon. This is diagrammed in figure 3.9.

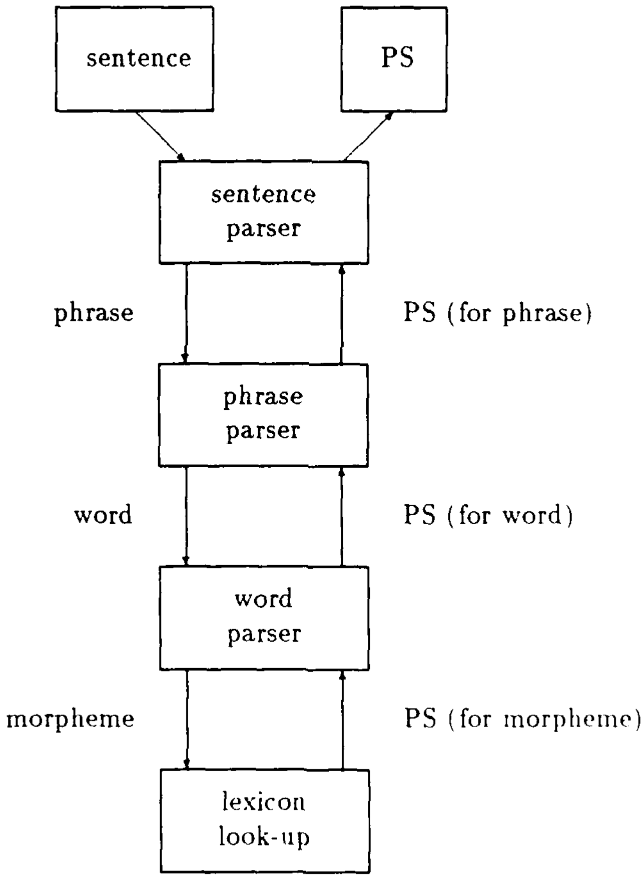


Figure 3.9: The recursion of the PS parser.

Each level of PS parsing uses the same engine. The basic algorithm is given in table 3.6. The first step of the algorithm is a loop that traverses the input from

left to right. Each unit (morpheme, word, or phrase) is sent to the subordinate parser for processing. The structure returned is then added to end of the PS for the current level. At this point unexecuted PS actions are tried to see if they can apply. To aid in the efficiency of processing, the parser employs an auxiliary structure, the set of unsatisfied predicates, that contains every node in PS that has at least one unexecuted action. Thus, step (b) of the main loop consists of a traversal of this set, attempting to execute each of the actions of the nodes within. Note that only the actions that pertain to the current level of parsing are considered.

1. Loop through constituents from left to right:
   a. call the subordinate parser, then
   b. execute applicable actions on adjacent trees.

2. Execute default actions.

3. Check well-formedness.

Table 3.6: The PS parsing engine.

PS actions are constrained to operate only on adjacent trees in the forest. That is, the actions in a node of one tree may only act on an adjacent tree. More specifically, because Warlpiri is a head-final language, PS actions may only apply to the preceding tree. Auxiliary processing, however, is a special case. The actions for auxiliary composition operate on the succeeding tree, rather than the preceding one.

The list of PS actions is given in table 3.7. The first routine concerns the interface to the syntactic processor. Most lexical items are relevant at the morphological level: for them `project-into-SS` will be a morphological action. When this action fires, the syntactic information of the lexical item is given to the syntactic parser which projects it according to its lexical information and enters it into SS. A link between the PS and syntactic nodes is kept for future processing.

The "normal" actions are used by the non-auxiliary elements. `Select` is the basic operation that causes the selector to become a sibling of its object (the tree to its left) in PS; at the same time, the syntactic counterpart of the object is declared to be an argument of the syntactic counterpart of the selector. This is the mechanism by which directed argument identification is performed. `Select*` is like `select` except that it is not deleted upon execution. This action is used to parse continuous case phrases. The last action, `inject`, is also similar to `select`, but the syntactic effect differs. Rather than declaring the object to be an argument of the injector, the syntactic information in the injector is added to (metaphorically speaking, injected into) the syntactic counterpart of the object. This action is used both by number-markers and tense elements for feature percolation.

The actions for processing the auxiliary word are quite like the other actions. `Right-adjacent` is used to build up the precedence structure of the auxiliary word.

interface to SS	`project-into-SS`

normal actions	`select (category)`
	`select* (category)`
	`inject (category)`

auxiliary actions	`auxiliary-adjacent ()`
	`right-adjacent (auxiliary categories)`
	`auxiliary-select (auxiliary category)`
	`auxiliary-inject (auxiliary category)`

Table 3.7: Available PS actions.

The argument of this action is a list of the categories of the auxiliary elements which may appear to the right of the node. For example, both auxiliary object clitics and dative clitics may appear to the right of the subject clitic, so each subject clitic would contain the lexical action in (5).

(5) `right-adjacent ({auxiliary-object, auxiliary-dative})`

The two actions, `auxiliary-select` and `auxiliary-inject` are analogous to `select` and `inject`. The difference is that these actions do not build up PS as do their counterparts; rather, they find their arguments in the set of siblings as constructed by `right-adjacent`. The auxiliary base selects its nominal agreement clitics, for example.

The last auxiliary action, `auxiliary-adjacent`, is used for auxiliary cliticization. The usual adjacency routine can not be used due to the strange nature of the auxiliary; instead, this special routine is used. `auxiliary-adjacent` may fire if there is something to its left, but no structure is built up. The purpose of this routine is to aid in checking the well-formedness of auxiliary cliticization.

The discussion of auxiliary parsing is best rounded out with an example. Consider the auxiliary word in (6), taken from the sample sentence. The computed entries for each element are shown in figure 3.10.

(6) `(KA RNA RLA)`

We start with the lexical parser. Following step one of the algorithm, the parser calls its subordinate, the morphological parser, with the first morpheme, *ka*. That parser computes its lexical entry, as shown above, and creates a node in PS for it. The second part of the main loop is then reached. There is only one morphological action, namely, `project-into-SS`. This action calls the syntactic processor with the auxiliary node as an argument. Following the `projections` action, the node projects two levels in SS. The `specifier` action also fires because there is a corresponding `license` action in the structure. At this point no more syntactic actions remain, so

```
(KA
  (ps (data (syllables 1))
      (actions
        (morphological (project-into-SS))
        (lexical (right-adjacent (auxiliary-subject
                                  auxiliary-object
                                  auxiliary-dative))
                 (auxiliary-select auxiliary-subject)
                 (auxiliary-select auxiliary-object)
                 (auxiliary-select auxiliary-dative))))
  (ss (data (tenses (nonpast))
            (aspect imperfect))
      (actions (projections 2)
               (license (argument . verb))
               (specifier (argument . verb)))))
(RNA
  (ps (data (syllables 1))
      (actions
        (morphological (project-into-SS))
        (lexical (auxiliary-adjacent)
                 (right-adjacent (auxiliary-object
                                  auxiliary-dative)))))
  (ss (data (person 1)
            (number singular))))
(RLA
  (ps (data (syllables 1))
      (actions (morphological (project-into-SS))
               (lexical (auxiliary-adjacent)))))
```

Figure 3.10: The computed entries for the elements of *ka-rna-rla*.

control is returned to the PS parser. As it turns out, there are no more morphological actions, so the morphological parse of *ka* is complete. The PS constructed so far is returned to the lexical parser and added to the lexical PS. The output structures at this point in the parse are displayed in figure 3.11.

<u>PS</u>

```
0: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                    AUXILIARY-SELECT: AUXILIARY-OBJECT
                    AUXILIARY-SELECT: AUXILIARY-SUBJECT
                    RIGHT-ADJACENT: (AUXILIARY-SUBJECT
                                     AUXILIARY-OBJECT
                                     AUXILIARY-DATIVE)
   data: SYLLABLES: 1
   morpheme: KA
   category: AUXILIARY-BASE
```

<u>SS</u>

```
projection?: NIL
category: AUXILIARY-BASE
children: projection?: T
          actions: ARGUMENT: VERB
          category: AUXILIARY-BASE
          children: projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE
```

Figure 3.11: The PS and SS after having parsed *ka*.

The lexical parser then executes the second step of the main loop. The only node with actions is the node for *ka*. However, none of its actions may execute, as there is no argument (*i.e.*, a node to its right) in PS. So, the loop is iterated, and the morphological parser is called for the second morpheme, *rna*. It is parsed similarly to *ka*, and a second node ultimately enters the lexical PS. When the second part of the main loop is reached again, there is indeed an action that may execute: the auxiliary base is combined with the subject clitic through firing the `right-adjacent` action. As a result of the previous action, the `auxiliary-select` action for the subject clitic may also fire. This causes the subject clitic to become an argument of the auxiliary base in a complement position, *i.e.*, as a sibling of the zero-level projection. The two structures at this point in the parse are shown in figure 3.12.

A key point here is how the linear template of the auxiliary morphemes is computed. This is effected through the `right-adjacent` action. Its argument consti-

PS

```
0: category: AUXILIARY-BASE
   children: 0: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                                 AUXILIARY-SELECT: AUXILIARY-OBJECT
                data: SYLLABLES: 1
                morpheme: KA
                category: AUXILIARY-BASE

             1: lexical actions: RIGHT-ADJACENT: (AUXILIARY-OBJECT
                                                  AUXILIARY-DATIVE)
                data: SYLLABLES: 1
                morpheme: RNA
                category: AUXILIARY-SUBJECT
```

SS

```
projection?: NIL
category: AUXILIARY-BASE
children: projection?: T
          actions: ARGUMENT: VERB
          category: AUXILIARY-BASE
          children: projection?: NIL
                    data: PERSON: 1
                          NUMBER: SINGULAR
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                    projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE
```

Figure 3.12: The PS and SS for *ka-rna*.

tutes a disjunction of the possibilities for the succeeding element in the string. The disjunction allows each component to be optional.

The other key to processing the auxiliary concerns syntactic processing. Syntactically, the base is considered the head of the auxiliary phrase, taking the agreement clitics as arguments. This is effected by the **auxiliary-select** action. Auxiliary bases contain one such action for each of the clitics, and every clitic that does appear in the input string is taken as an argument of the base. Note that this form of selection can not be folded into the adjacency action as with non-auxiliary components because one element effects the adjacency while another (the base) effects the syntactic selection.

The last morpheme of the input word, *rla*, is parsed just like the others. Once its morphological PS enters the lexical PS, it is made adjacent to the auxiliary word via the **right-adjacent** action of *rna*, and selected by the base, *ka*. The final output structures for this auxiliary word are shown in figure 3.13.

Returning to the PS parsing algorithm, we come to the second step: perform default actions. Currently there is only one default PS action that inserts the phonologically null absolutive case-marker. After the phrasal parse is complete, the default action checks to see if the phrase ends with a noun that has not been inflected for case.[6] In this event, a node for the absolutive case-marker is inserted after the last noun, and the PS parser is called once again to execute the actions of the newly inserted node. Consider the absolutive case phrase in (7).

(7) ((YIRRINJI) (YIRRARU) (KARDIRRPA))

After the main phrasal loop has completed, the default action will detect that the last word, *kardirrpa*, has not been inflected for case. Accordingly, it will insert a node for the absolutive case-marker, whose lexical information is shown in figure 3.14.[7]

The default action executes the morphological and lexical actions directly, and then reinvokes the PS parser to process the modified phrase. The morphological action causes the case-marker to project into SS, and the lexical action causes the case-marker to select the right-most noun, just as overt case-markers would. The phrasal action, **select***, selects the preceding nouns, again, just as overt case-markers would. Figure 3.15 shows the results.

The last operation of the engine is to check parse well-formedness.[8] The nature of the check depends on the phonological level. At the morphological level, as one might expect, there are no conditions. The remaining levels, however, do examine the precedence structure.

At the lexical level, three checks are performed. The first makes sure that the PS consists of one tree, ensuring that every element in the word can be adjoined to

[6]This action takes care to ignore the auxiliary word which may be the last of the phrase; if present, the second-to-last word is checked.

[7]The absolutive case-marker is not stored in the lexicon *per se*. Instead, the node exists as the value of a special variable that has been set to contain the data and actions of the case-marker

[8]In the event of an error, the parser immediately halts and returns an error message, along with the output structures extant at the time of the detection. Thus, the parser takes any illformedness to be fatal.

PS

```
0: category: AUXILIARY-BASE
   children: 0: lexical actions: AUXILIARY-SELECT: AUXILIARY-OBJECT
                data: SYLLABLES: 1
                morpheme: KA
                category: AUXILIARY-BASE

             1: data: SYLLABLES: 1
                morpheme: RNA
                category: AUXILIARY-SUBJECT

             2: data: SYLLABLES: 1
                morpheme: RLA
                category: AUXILIARY-DATIVE
```

SS

```
projection?: NIL
category: AUXILIARY-BASE
children: projection?: T
          actions: ARGUMENT: VERB
          category: AUXILIARY-BASE
          children: projection?: NIL
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

                    projection?: NIL
                    data: PERSON: 1
                          NUMBER: SINGULAR
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                    projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE
```

Figure 3.13: The PS and SS for *ka-rna-rla*.

```
(ps (actions (morphological (project-into-SS))
             (lexical (select noun))
             (phrasal (select* noun))))

(ss (data (case-marked absolutive))
    (actions (projections 1)))
```

Figure 3.14: The lexical information for the absolutive case-marker.

some other part. There is one exception that allows the second, unconnected word to be an auxiliary word. As mentioned above, auxiliaries are not entered into PS like other words, so this case must be allowed. Auxiliary well-formedness is covered later.

The second check ensures that the word (and the optional auxiliary word) contains at least two syllables. In fact, this check need not be explicit for nouns and verbs. In Warlpiri there are no single-syllable nouns, so all nouns, however they are inflected, will pass. Verbs must always be inflected for tense, and since there are no null tense elements, they too will always pass this test. Hence the check remains for auxiliary words, where the syllables of each element are summed and compared to two. This is why there is no explicit syllable information for non-auxiliary lexical items.

The last lexical check makes sure that all clitics are, in fact, enclitic to something. This is implemented by examining the lexical actions for the left-most node of the word. If it contains an unexecuted `select` or `inject` action, then it is an unsatisfied clitic, and flagged as such. This check is also performed for the auxiliary word: if it contains an `auxiliary-adjacent` action, the word is declared ungrammatical and an error is signalled.

At the phrasal level only one check is performed. Like the lexical level, phrases are required to consist of one tree (again, with the possible exception of a trailing auxiliary word). This condition stems from the fact that phonological phrases may not contain more than one syntactic phrase.

At the sentential level auxiliary positioning is checked. The test is simple: the auxiliary word, if present, must be either in the first or second position. Of course, this not the entire condition, as some auxiliaries are required to be in the second position. But this requirement is taken care of by the cliticization check: only those auxiliary words that must be enclitic to something are required to be in second position.

This completes discussion of the main loop. When the sentential parser has completed, the syntactic default actions are executed, followed by the syntactic well-formedness checks. After the syntactic processing, the semantic default actions and well-formedness checks are called. Once this point is reached, the entire parse is finished.

PS

```
0: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: YIRRINJI
                category: NOUN

             1: category: CASE
                children: 0: morpheme: YIRRARU
                             category: NOUN

                          1: category: CASE
                             children: 0: morpheme: KARDIRRPA
                                          category: NOUN

                                       1: morpheme: *ABS*
                                          category: CASE
```

SS

```
projection?: NIL
category: CASE
children: projection?: NIL
          morpheme: YIRRINJI
          category: NOUN

          projection?: NIL
          morpheme: YIRRARU
          category: NOUN

          projection?: NIL
          morpheme: KARDIRRPA
          category: NOUN

          projection?: T
          data: CASE-MARKED: ABSOLUTIVE
          morpheme: *ABS*
          category: CASE
```

Figure 3.15: The PS and SS for (7).

Parsing the Precedence Phenomena

In the introduction to this chapter I listed the phenomena for which the parser is responsible; see table 3.1. Under the listing for precedence phenomena there were three areas: composition (for nouns, verbs, and auxiliaries), continuous case phrases, and auxiliary positioning. In this section I discuss how the PS parser handles these phenomena.

Of the three categories of word composition, the auxiliary has already been discussed. Nominal and verbal composition are quite similar, so only an example of the latter will be given. Consider the verb in (8).

(8) `(PUNTA RNI)`

As with the auxiliary example above, we start the parser at the lexical level. In its main loop it calls the morphological parser to process the verb stem, *punta*. The PS and SS for *punta* were given in figure 3.7. As no actions may fire, the lexical loop iterates, and the tense element, *rni*, is parsed. Its computed lexical entry is given in figure 3.16, and the resulting structures are given in figure 3.17. Note that this element does not project into SS on its own.

```
(RNI
  (ps (actions (lexical (inject verb))))
  (ss (data (tense nonpast))
      (actions (license (case-assign . absolutive)))))
```

Figure 3.16: The computed entry for *rni*.

PS

```
0: lexical actions: INJECT: VERB
   morpheme: RNI
   category: TENSE
```

SS

Figure 3.17: PS and SS for *rni*.

When the morphological PS is entered into the lexical PS, the `inject` operation of the tense element fires. In PS, this causes the verb stem and the tense element to become siblings under a single tree. In SS, the syntactic information of *rni* is added to that of *punta*. Once the `license` action joins the other actions of the V node, it acts in concert with the remaining `complement` action so as to place the `case-assign` action for the absolutive case in the zero-level projection of the verb. The resulting structures are shown in figure 3.18.

<u>PS</u>

```
0: category: VERB
   children: 0: data: CONJUGATION: 2
                morpheme: PUNTA
                category: VERB

             1: morpheme: RNI
                category: TENSE
```

<u>SS</u>

```
projection?: NIL
category: VERB
children: projection?: T
          actions: CASE-ASSIGN: ERGATIVE
                   THETA-ASSIGN: AGENT
          category: VERB
          children: projection?: T
                    actions: CASE-ASSIGN: ABSOLUTIVE
                             THETA-LINK: PATH
                             THETA-ASSIGN: THEME
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME PATH)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          PATH: DATIVE
                          SUBJECT: AGENT
                          OBJECT: PATH
                    morpheme: PUNTA
                    category: VERB
```

Figure 3.18: The PS and SS for *punta-rni.*

The second item on the list of phenomena concerns the processing of continuous case phrases. This was demonstrated in the discussion of the absolutive case-marker, above. So, the remaining phenomenon to be presented is auxiliary positioning.

The auxiliary appears in several different forms. It may be a clitic, a word unto itself, or in a phrase by itself. It may be in either the first or second position. But not all manifestations of the auxiliary may appear in all of these positions. Therefore a demonstration of the parser's ability to handle this phenomenon would require an exhaustive test. Here I present an indicative example; more tests can be found in the appendix. Consider (9) which is a minor variation of the sample sentence, in which the auxiliary base, *ka*, has been removed. The equivalent input representation is given in (10).

(9) *Ngajulu-rlu-rna-rla punta-rni kurdu-ku karli.*
I-ERG-1s-3d take-NONPAST child-DAT boomerang
'I will take the boomerang from the child.'

(10) `(((NGAJULU RLU RNA RLA)) ((PUNTA RNI)) ((KURDU KU)) ((KARLI)))`

Because the auxiliary word begins with a nominal agreement clitic, it must be enclitic to the preceding word, and therefore in second position. The PS for this sentence is given in figure 3.19. The key to the simplicity of the well-formedness check is the action `auxiliary-adjacent`. This action, and the lexical well-formedness check that the action has fired, guarantee cliticization when necessary, thus ruling out certain auxiliaries from appearing in first position. Furthermore, since auxiliary adjacency does not combine the auxiliary with other trees in PS (at either the lexical, phrasal, or sentential levels), the check for the second position simply consists of looking at the second tree.

3.2.2 The Syntactic Parser

As mentioned above, the syntactic parser is called whenever a unit of PS becomes syntactically relevant. The first part of this section describes this mechanism from the point of view of the syntactic parser. Afterwards, the main parsing algorithm is given. Finally, the algorithm is demonstrated on the syntactic phenomena handled by the parser.

Syntactic Relevance

There are three ways in which a unit of PS may become syntactically relevant: projection into SS, selection, and injection. The main mechanism is projection into syntax, performed by the action `project-into-SS`. This routine first projects the item according to its lexical information and then adds the tree to SS.

The second mechanism is selection, which is used both by auxiliary and non-auxiliary elements. This action is used when the ordering of one element with respect to another has a syntactic effect. Specifically, the selector is taken to be the predicator and the selected is taken to be one of its arguments. For example,

PS

```
0: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: NGAJULU
                category: NOUN

             1: morpheme: RLU
                category: CASE

1: category: AUXILIARY-SUBJECT
   children: 0: data: SYLLABLES: 1
                morpheme: RNA
                category: AUXILIARY-SUBJECT

             1: data: SYLLABLES: 1
                morpheme: RLA
                category: AUXILIARY-DATIVE

2: category: VERB
   children: 0: data: CONJUGATION: 2
                morpheme: PUNTA
                category: VERB

             1: morpheme: RNI
                category: TENSE

3: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: KURDU
                category: NOUN

             1: morpheme: KU
                category: CASE

4: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: KARLI
                category: NOUN

             1: morpheme: *ABS*
                category: CASE
```

Figure 3.19: PS for (10).

case-markers select nouns. This action causes the syntactic counterpart of the noun to be an argument of the corresponding case phrase. In structural terms, the noun is made the sibling of the zero-level projection of the case-marker.

Injection is the last channel from PS to SS. With this action the syntactic information of the injector is added to that of the injected. An example of this concerns the parsing of inflected verbs. When the tense element cliticizes to the verb stem, its syntactic information is added to that of the verb's. The licensing action for the case-marking of absolutive case is the main bit of information that is added. The syntactic parser will then be able to add the case-marking action to verbal projection (in the zero-level node), because both the `complement` action and the `license` action for the case-marking action are in place.

After any of these actions has executed, the syntactic parser is called to see if any further syntactic actions can apply, as a result of the addition of syntactic structure or information. The next section describes the workhorse of the syntactic parser.

The Basic Engine

The parsing algorithm, given in table 3.8, loops through the unexecuted actions of the syntactic nodes to see if any may be executed. (Like the PS engine, this parser also employs a set of unsatisfied predicates for efficiency's sake.) Note that previously existing actions are also checked because the newly added structure may provide arguments appropriate for them.

1. Loop over every unexecuted action in SS:
   if the action applies either to a sibling or to a tree
   in the structure, apply the action.
2. If any actions have fired, try the above loop again.

Table 3.8: The syntactic parsing engine.

There are four syntactic actions available, as listed in table 3.9. The first action, `case-assign`, performs the second half of case assignment. The first step, case-marking, need not be an action, since that feature is inherent in all of the case-markers. That is, the SS datum, `case-marked`, is stored in the lexical entry for each of the case-markers (for their own case, of course). The next two actions are for θ-assignment. The second such action, `theta-link`, is needed only for the path θ-role, as it is assigned directly by the dative case-marker. The last action is for general argument taking. Currently, it is used only for the auxiliary which takes the verbal projection as its sole argument.

We now return to the second step of the parsing algorithm. If any actions are executed in the course of traversing the set of unsatisfied predicates, the loop is tried once again. In this way, all the actions that are applicable due to the introduction of new syntactic structure will be executed. This loop also eliminates any ordering effects of the actions in that if one action is dependent on another, it doesn't matter

```
case-assign (case)
theta-assign (role)
theta-link (role)
argument (category)
```

Table 3.9: Available syntactic actions.

which way they're listed: both will be executed. Of course, this looping does not mitigate the ordering effect of two actions that may be applicable to the same node at the same time. The parser assumes that at most one such action will be applicable.[9]

As with the PS parser, there are also routines for performing default actions and checking the well-formedness of the syntactic structure. Both of these routines are called by the PS parser once it has completed the sentential level of processing.

The only default action of the syntactic parser is to supply an auxiliary word if one is not present in the input string. Such an auxiliary consists of the null base and null agreement clitics, which contain the default information as mentioned above. Note that placing the handler for the zero auxiliary in syntax eliminates the need for the parser to guess where the auxiliary is to be placed in the input sentence; its placement does not matter, only its syntactic (and then semantic) effects concern the parser.

The well-formedness check for SS consists of making sure that the structure contains exactly one tree. This check subsumes the Case Filter in that nouns will not get linked into the verbal projection unless they are appropriately marked for case. Nouns that are not marked for case will remain as separate trees in the structure, and will thus be flagged as in error by this routine.

Parsing the Syntactic Phenomena

The parser can handle two types of syntactic phenomena: determination of grammatical function and free phrase order. In this section I discuss how the syntactic parser goes about this.

The majority of the work in determining grammatical function is actually performed in the execution of the lexical rules, as described above. The syntactic parser just makes them stick by allowing actions to operate only on siblings, and not on other parts of the structure. In this way, the actions placed in the structure at their appropriate levels will, in fact, cause their arguments to be placed in the proper places. Subjects will be adjoined as siblings of the first-level projection, and objects will be adjoined next to the zero-level projection.

Consider the verb stem *punta* once again. Given its θ-grid, and the lexical rules above, the parser determines the mapping from case and θ-role to grammatical function, as shown in table 3.10. This mapping is implemented by placing the case-

[9]This method assumes unambiguous syntactic structures. Of course, there is structural ambiguity in Warlpiri—as in any other natural language. This shortcoming of the parser is discussed in the concluding chapter.

and θ-actions at their respective levels in the projection. Figure 3.18, the SS for the inflected verb, demonstrated this placement.

case	θ-role	grammatical function
ergative	agent	subject
absolutive	theme	object
dative	path	object

Table 3.10: The case/θ/GF mapping for *punta*.

The key to processing the second phenomenon, free phrase order, lies in the lack of precedence information in SS. When the syntactic engine searches for potential arguments, it traverses the entire set of trees, regardless of the order in which they were added. The search also involves previously adjoined trees, *i.e.*, siblings of the node containing the action to be executed. Siblings must be checked because more than one action may apply to a single node.

Consider a permutation of the sample sentence, shown in (11). In order to appreciate the syntactic parser's indifference to precedence in the input string, we focus on the entry of the inflected verb into SS. When it projects into SS its argument-taking actions (*e.g.*, `case-assign`) become manifest in its projection. The main loop of the syntactic engine then starts up, searching for arguments. It finds all of the arguments that are present and joins them to the verbal projection; the as yet unentered actions are not joined, and the actions performing their adjunction are simply left unexecuted.

(11) (((KARLI)) ((KA RNA RLA)) ((PUNTA RNI)) ((NGAJULU RLU))
((KURDU KU)))

In (11) the verb enters after one of its arguments, *karli*, has already projected into SS. Therefore, only the actions for the object may fire. The intermediate SS is shown in figure 3.20; the syntactic structure of the auxiliary has been omitted for brevity. When the fourth word, *ngajulu-rlu*, enters, the subject actions fire: and when the last word projects into SS, the indirect object actions fire, completing the parse.

3.2.3 Semantic Interpretation

As mentioned above, semantic processing is performed on SS. After the syntactic structure has been checked for well-formedness, it is checked for semantic well-formedness. These checks are described below, but first I begin with the major semantic operation, argument identification.

SS

```
projection?: NIL
data: ARGUMENT: VERB
category: VERB
children: projection?: T
          actions: CASE-ASSIGN: ERGATIVE
                   THETA-ASSIGN: AGENT
          category: VERB
          children: projection?: NIL
                    data: THETA-ASSIGNED: THEME
                          CASE-ASSIGNED: ABSOLUTIVE
                    category: CASE
                    children: projection?: NIL
                              morpheme: KARLI
                              category: NOUN

                              projection?: T
                              data: CASE-MARKED: ABSOLUTIVE
                              morpheme: *ABS*
                              category: CASE

                    projection?: T
                    actions: THETA-LINK: PATH
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME PATH)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          PATH: DATIVE
                          SUBJECT: AGENT
                          OBJECT: PATH
                    morpheme: PUNTA
                    category: VERB
```

Figure 3.20: The intermediate SS for (11).

Argument Identification

Argument identification is the process of associating nouns with the semantic roles they fill. The algorithm for argument identification is fairly simple. For each role in the verb's θ-grid, the verbal projection is searched for the nouns that have been assigned that role. For example, the mapping of θ-roles to nouns for the sample sentence, (1), is given in figure 3.21.

```
PATH: projection?: NIL
      morpheme: KURDU
      category: NOUN

THEME: projection?: NIL
       morpheme: KARLI
       category: NOUN

AGENT: projection?: NIL
       data: PERSON: 1
             NUMBER: (SINGULAR)
       morpheme: NGAJULU
       category: NOUN
```

Figure 3.21: The mapping of θ-roles to case phrases for (1).

Well-Formedness Checks

Two semantic well-formedness checks are performed by the parser, each dealing with a different part of the auxiliary. The first ensures that the auxiliary base is appropriate for the tense of the inflected verb. This check is rather straightforward: the sentence is well-formed if the tense of the tense element is a member of the set of the allowable tenses of the base. For example, the sample sentence passes this test because its tense, non-past, is allowed by the base, *ka*. If the the other imperfective base, *-lpa*, were used it would be ill-formed.

The second well-formedness check concerns the agreement clitics, and consists itself of two parts. The first part checks licensing of clitics. That is, agreement clitics are grammatical only if there are corresponding arguments in the sentence, as licensed by the verb. Consider the verb stem, *punta*. It selects three θ-roles which are manifest as subject, object, and indirect object; all three arguments are licensed. By the rules concerning auxiliary registration, the subject and the indirect object (*i.e.*, the ergative and dative case phrases, respectively) must be registered in the auxiliary with clitics (which may not be phonologically overt, however). On the other hand, consider the verb stem, *nyina* 'to be; to sit'. It selects a single θ-role which shows up as the subject. As a result, the auxiliary may not appear with an object clitic or a dative registration marker.

The other part of the agreement check matches person and number information of the clitics and the arguments. This check is also simple: the persons and numbers must agree.[10] There is a slight twist with nominals with no overt number-marker: they match either singular or plural number.

There is another twist in this processing, namely, handling null clitics and null anaphora. Null clitics have default values of third-person and singular. In the event of a null clitic, these values are retrieved and matched as usual. Null anaphora are handled differently. Given the intrasentential processing of the parser, these elements are considered as wild-cards for the purposes of agreement; any clitic will match them. In a parser that handles more than one sentence, the clitics will have to match their referents, just as in the case of overt arguments.

Demonstrating the parser's performance on these phenomena is best done with an exhaustive test of the possibilities. Such a list can be found in the appendix.

3.3 Parsing the Sample Sentence

Below is a parse trace of the sample sentence, (1). Each line corresponds to the execution of a single action. Lines begin with the name of the node performing the action, followed by the phonological level at which the action took place (for PS actions only). Lines end with the action that executed and its arguments.

The first five actions parsed the first word, *ngajulu-rlu*. *Ngajulu* projected into SS, and the syntactic parser then projected it according to its lexical information, which was for zero levels. *Rlu* was then parsed similarly. Once both morphemes entered PS, they were combined with the `select` action of *rlu*.

```
NGAJULU (MORPHOLOGICAL): PROJECT-INTO-SS()
NGAJULU: PROJECTIONS(0)
RLU (MORPHOLOGICAL): PROJECT-INTO-SS()
RLU: PROJECTIONS(1)
RLU (LEXICAL): SELECT(NOUN)
```

The next set of actions parsed the auxiliary, *ka-rna-rla*. First, *ka* projected into SS, projected two levels, and placed its `argument` action (which later executed on the verbal projection) in its specifier position. Then the subject clitic, *rna*, entered and also projected into SS. Once *rna* entered PS, *ka* was able to first combine with it and then select it, with `right-adjacent` and `auxiliary-select`, respectively. The dative registration clitic, *rla*, was then parsed similarly to the subject clitic.

```
KA (MORPHOLOGICAL): PROJECT-INTO-SS()
KA: PROJECTIONS(2)
KA: SPECIFIER((ARGUMENT . VERB))
RNA (MORPHOLOGICAL): PROJECT-INTO-SS()
```

[10] This formulation is not quite right for a full account of Warlpiri agreement, but it serves for the simple range of the parser. See [Nas86] for a more nearly complete description of this intricate phenomenon.

```
RNA: PROJECTIONS(0)
RNA (LEXICAL): AUXILIARY-ADJACENT()
KA (LEXICAL): RIGHT-ADJACENT((AUXILIARY-SUBJECT AUXILIARY-OBJECT
                              AUXILIARY-DATIVE))
KA (LEXICAL): AUXILIARY-SELECT(AUXILIARY-SUBJECT)
RLA (MORPHOLOGICAL): PROJECT-INTO-SS()
RLA: PROJECTIONS(0)
RLA (LEXICAL): AUXILIARY-ADJACENT()
RNA (LEXICAL): RIGHT-ADJACENT((AUXILIARY-OBJECT AUXILIARY-DATIVE))
KA (LEXICAL): AUXILIARY-SELECT(AUXILIARY-DATIVE)
```

The following set of actions parsed the verb, *punta-rni.* First the verb stem, *punta*, projected into SS. Once the verbal projection entered SS, the auxiliary (headed by *ka*) was able to attach it in its specifier position, by the **argument** action. The next actions see the verb placing its case- and θ-actions in its structure. Note that the assignment of ergative case and the agent θ-role execute now because the ergatively marked noun, *ngajulu-rlu*, is present in SS. The other case- and θ-actions of the verb must wait until the arguments appear in SS. The last action for parsing the verb concerns the tense element, *rni*, which combines with the verb stem and injects its syntactic information into the verbal projection. (The added information of interest is the action licensing of absolutive case assignment.)

```
PUNTA (MORPHOLOGICAL): PROJECT-INTO-SS()
PUNTA: PROJECTIONS(2)
PUNTA: SPECIFIER((THETA-ASSIGN . AGENT))
KA: ARGUMENT(VERB)
PUNTA: SPECIFIER((CASE-ASSIGN . ERGATIVE))
PUNTA: CASE-ASSIGN(ERGATIVE)
PUNTA: THETA-ASSIGN(AGENT)
PUNTA: COMPLEMENT((THETA-ASSIGN . THEME))
PUNTA: COMPLEMENT((THETA-LINK . PATH))
PUNTA: COMPLEMENT((CASE-ASSIGN . ABSOLUTIVE))
RNI (LEXICAL): INJECT(VERB)
```

The fourth word, *kurdu-ku*, was parsed with the executed actions below. The parse here proceeded quite like that of the first word, *ngajulu-rlu*. Note that once the dative case phrase entered SS it was linked to the verbal projection via the **theta-link** action of the verb.

```
KURDU (MORPHOLOGICAL): PROJECT-INTO-SS()
KURDU: PROJECTIONS(0)
KU (MORPHOLOGICAL): PROJECT-INTO-SS()
KU: PROJECTIONS(1)
PUNTA: THETA-LINK(PATH)
KU (LEXICAL): SELECT(NOUN)
```

The last sequence of actions shows the parse of *karli.* Here we see the default actions execute at the phrasal level. The absolutive case-marker (shown here as ***ABS***) entered into PS at the phrasal level. It then projected into SS, as would other case-markers. Immediately, the case phrase was attached as an object of the verbal projection, via the **case-assign** and **theta-assign** actions. Finally, the noun, *karli*, was selected by the case-marker.

```
KARLI (MORPHOLOGICAL): PROJECT-INTO-SS()
KARLI: PROJECTIONS(0)
*ABS* (PHRASAL): PROJECT-INTO-SS()
*ABS*: PROJECTIONS(1)
PUNTA: CASE-ASSIGN(ABSOLUTIVE)
PUNTA: THETA-ASSIGN(THEME)
*ABS* (PHRASAL): SELECT(NOUN)
```

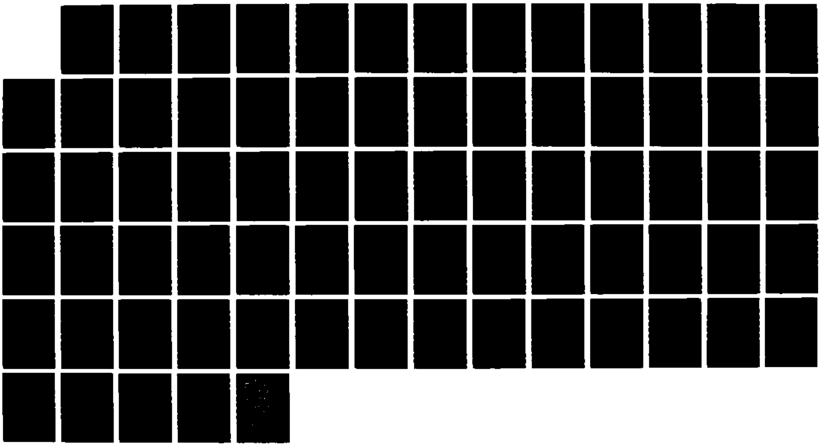
This completes the parsing trace of the sample sentence. Below I show the resulting PS and SS for this sentence. The interpretation of the SS, *i.e.*, the mapping of the verb's θ-roles to words, was given in figure 3.21.

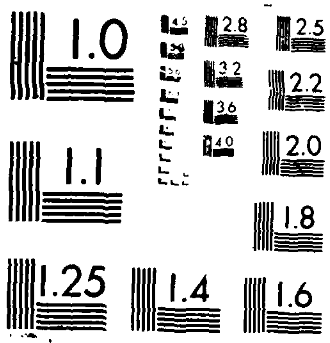
The precedence structure for sentence (1):

```
0: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: NGAJULU
                category: NOUN

             1: morpheme: RLU
                category: CASE

1: category: AUXILIARY-SUBJECT
   children: 0: lexical actions: AUXILIARY-SELECT: AUXILIARY-OBJECT
                data: SYLLABLES: 1
                morpheme: KA
                category: AUXILIARY-BASE

             1: data: SYLLABLES: 1
                morpheme: RNA
                category: AUXILIARY-SUBJECT

             2: data: SYLLABLES: 1
                morpheme: RLA
                category: AUXILIARY-DATIVE

2: category: VERB
   children: 0: data: CONJUGATION: 2
                morpheme: PUNTA
```

AD-A189 381 A GOVERNMENT-BINDING BASED PARSER FOR WARLPIRI A FREE-WORD ORDER LANGUAGE..(U) MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE L.. M B KASHKET 2/2

UNCLASSIFIED 22 SEP 87 AI-TR-993 N00014-85-K-0124 F/G 5/7 NL




1.0
2.8
2.5
3.2
2.2
3.6
4.0
2.0
1.1
1.8
1.25
1.4
1.6

```
            category: VERB

         1: data: CONJUGATION: 2
            morpheme: RNI
            category: TENSE

3: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: KURDU
                category: NOUN

             1: morpheme: KU
                category: CASE

4: phrasal actions: SELECT*: NOUN
   category: CASE
   children: 0: morpheme: KARLI
                category: NOUN

             1: morpheme: *ABS*
                category: CASE
```

The syntactic structure for sentence (1):

```
projection?: NIL
category: AUXILIARY-BASE
children: projection?: NIL
          data: ARGUMENT: VERB
          category: VERB
          children: projection?: NIL
                    data: THETA-ASSIGNED: AGENT
                          CASE-ASSIGNED: ERGATIVE
                    category: CASE
                    children: projection?: NIL
                              data: PERSON: 1
                                    NUMBER: (SINGULAR)
                              morpheme: NGAJULU
                              category: NOUN

                              projection?: T
                              data: CASE-MARKED: ERGATIVE
                              morpheme: RLU
                              category: CASE

                    projection?: T
                    category: VERB
```

```
          children: projection?: NIL
                    data: THETA-ASSIGNED: THEME
                          CASE-ASSIGNED: ABSOLUTIVE
                    category: CASE
                    children: projection?: NIL
                              morpheme: KARLI
                              category: NOUN

                              projection?: T
                              data: CASE-MARKED: ABSOLUTIVE
                              morpheme: *ABS*
                              category: CASE

                    projection?: NIL
                    data: THETA-LINKED: PATH
                    category: CASE
                    children: projection?: NIL
                              morpheme: KURDU
                              category: NOUN

                              projection?: T
                              data: CASE-ASSIGNED: DATIVE
                                    CASE-MARKED: DATIVE
                                    THETA-ASSIGNED: PATH
                              morpheme: KU
                              category: CASE

                    projection?: T
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME PATH)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          PATH: DATIVE
                          SUBJECT: AGENT
                          OBJECT: PATH
                    morpheme: PUNTA
                    category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
          morpheme: RLA
          category: AUXILIARY-DATIVE

          projection?: NIL
```

```
data: PERSON: 1
      NUMBER: SINGULAR
morpheme: RNA
category: AUXILIARY-SUBJECT

projection?: T
data: TENSES: (NONPAST)
      ASPECT: IMPERFECT
morpheme: KA
category: AUXILIARY-BASE
```

Chapter 4

Conclusion

The first section of this chapter discusses some other current grammatical frameworks and how they address to phenomenon of free word order. Following this it turns to the parser's shortcomings and how they might be overcome.

The parser presented here is not the only member of its family, but it has only a few cousins. The closest relative is another Government-Binding based Warlpiri parser, written by Brunson-Loker. Unfortunately, the work has not yet been published, and I have not yet had a chance to view the system in operation, so I can not comment on it here. However, a comparison of the parsers should prove to be quite interesting.

There are also a few GB-based processors in the literature that work on languages other than Warlpiri. A comparison of these works is beyond the scope of this thesis, but I present the list of the systems of which I know in table 4.1. The interested reader is referred to the original publications.

- Abney's English parser[Abn87]
- Dorr's English-Spanish translator[Dor87]
- Sharp's English-Spanish translator[Sha85]
- Wehrli's French parser[Weh84]

Table 4.1: Other GB-based processors.

4.1 Other Grammatical Frameworks

There are many grammatical frameworks besides Government-Binding theory. However, I shall only be able to discuss a few of them here. In particular, I will discuss ID/LP Grammar, Lexical-Functional Grammar, and Tree-Adjoining Grammar. Of course, these reviews are brief, and therefore do not do justice to the entire content of the theories; this discussion focuses only on their ability to analyze the phenomenon of free word order.

4.1.1 ID/LP Grammar

ID/LP grammars[Cur61] contain two kinds of rules. ID rules dictate the *immediate dominance* relations of the constituents of the grammar, while LP rules constrain the *linear precedence* among the children of a parent node. Such a grammar was actually presented in the introduction, under the guise of a modified context-free grammar. That grammar for simple transitive sentences is repeated here as (1).

(1) $S \rightarrow \{NP_s, VP\}$
$VP \rightarrow \{V, NP_o\}$

These are the two ID rules of the grammar. The first rule states that an S consists of an NP and a VP in either order; the second states that a VP consists of a V and an NP, also in either order. This grammar does not contain any LP rules, however.

As mentioned in the introduction, grammars of this sort can suffer from inadequate coverage. For example, the ID/LP grammar above can not generate either of the sentence schemata found in (2).

(2) $V\ NP_s\ NP_o$
$NP_o\ NP_s\ V$

This poverty of coverage seems to be at odds with the structures of the parser presented here. After all, one of its structures represents precedence and one represents hierarchy, which are very similar to the LP and ID rules, respectively. The key difference, however, lies in the *scope* of the hierarchical relations. In ID/LP grammars, all the children of a single sibling must be adjacent to each other, subject to the linear precedence given in the LP rules. For the parser, on the other hand, there is no such restriction on the children of a syntactic node; they may appear anywhere in the input string. Thus, the languages allowed by the parser's grammar subsume ID/LP languages, and allow for the scrambling evident in free-word order languages. It is important to note that this coverage is not gained at the expense of linguistic perspicuity: the parser is still able to represent the relevant hierarchical structure, in order to recover semantic roles.

4.1.2 Lexical-Functional Grammar

Klavans[Kla82] has shown that Lexical-Functional Grammar[Bre82] can account for free-word order phenomena in much the same way as presented here. The key to analyzing the similarity between free- and fixed-order systems lies in LFG's bipartite representation of *c-structure* and *f-structure*. C-structure is ordered by precedence and hierarchy, and is used to represent the ordered phenomena, such as continuous case phrases. F-structure, on the other hand, is not ordered by precedence, rather it is strictly a hierarchical structure, used to represent grammatical functions. Like the GB grammar used by the parser, c-structure is related to f-structure by case-marking.

The LFG analysis can be illustrated for a simple language of transitive sentences. The grammar given here, taken from Klavans' paper, is for Ngiyambaa, another aboriginal language from Australia that is quite similar to Warlpiri. The c-structure rules are shown in (3).[1]

(3) $S \rightarrow \alpha$ (Encl) α^* (where $\alpha = X, \overline{X}$)
$\overline{N} \rightarrow N^*$
$\overline{V} \rightarrow V^*$

These rules cover continuous case phrases, labeled '$\overline{N}$'. Single-noun case phrases are also handled; they are labeled 'N'. The structures derived with c-structure are annotated both with grammatical function and case, as shown in (4), which gives the annotations both for the subject and the object. The key here is that each noun or noun phrase marked for a given case will be annotated with the same grammatical function. Thus, the position of the noun or noun phrase does not matter; only its case-marking is involved in the determination of its grammatical role.

(4) a. (SUBJ)
(CASE = ERG)
b. (OBJ)
(CASE = ABS)

Annotated c-structure elements are mapped into f-structure by their grammatical function. The f-structure schema for transitive verbs is shown in figure 4.1. The 'PRED' slots in the structure are used to hold the lexical items that correspond to grammatical functions indicated by the 'SUBJ' and 'OBJ' slots. Like the lexical theory used by the parser, LFG states that the subcategorization for arguments is dictated by the predicator; this information is shown in the top-level 'PRED' slot. In passing, note that LFG takes grammatical functions to be elementary and so the subcategorization is stated in those terms; GB takes these functions to be derivative from semantic roles, and so states predicative selection in term of θ-roles.

$$\left| \begin{array}{ll} \text{SUBJ} & \left| \begin{array}{ll} \text{CASE} & \text{ERG} \\ \text{PRED} & x \end{array} \right| \\ \\ \text{OBJ} & \left| \begin{array}{ll} \text{CASE} & \text{ABS} \\ \text{PRED} & y \end{array} \right| \\ \\ \text{PRED} & \left| \, v \ \ \langle \text{ SUBJ, OBJ } \rangle \right| \end{array} \right|$$

Figure 4.1: The f-structure schema for transitive verbs.

Thus we see that LFG functions very similarly to the GB account proposed in this thesis. The separation of precedence and hierarchy is the key to handling both

[1]The category 'Encl' (enclitic) is used for the positioning of the auxiliary (which must also appear in Wackernagel's position, roughly speaking).

free- and fixed-order phenomena. While there are other theoretical differences that prevent a merging of the two linguistic camps, both LFG and GB seem to be on the same footing here. It is unfortunate that there has been no published work on LFG-based parsing of free-word order phenomena.

4.1.3 Tree-Adjoining Grammar

A recent development in the framework of Tree-Adjoining Grammar[Jos] (TAG) presents an interesting account of free-word order phenomena. TAG(LD/LP) provides *local domination* structures over which *linear precedence* can be defined. LD structures can be thought of as ID rules that may be more than one level deep; ID rules are equivalent to LD structures of depth one. For example, the equivalent of the ID rules in (1) are shown in figure 4.2. TAG(LD/LP) extends this notion by allowing structures of arbitrary depth.

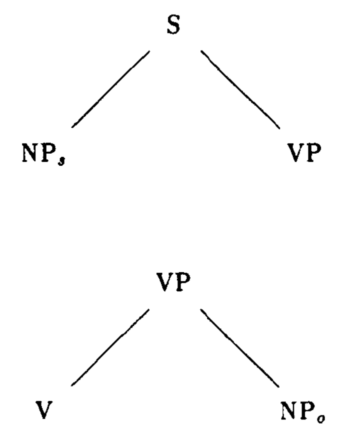


Figure 4.2: The structural equivalents of the ID rules in (1).

A TAG(LD/LP) that covers the language of transitive sentences is shown in figure 4.3. Note that the grammar includes only a single domination structure, and no linear precedence relations. This grammar does, indeed, generate the six permutations of the language. Furthermore, it represents the hierarchical relations that are linguistically motivated.

However, there is a problem with TAG(LD/LP): the grammar has too great an expressive power. There are no constraints on the composition of the domination structures, so structures of arbitrary size can be used. This freedom allows TAG(LD/LP) to escape the limits of ID/LP and represent free word order (at least as far as the sample language demonstrates). But TAG(LD/LP) seems to be too general, and thereby lose its explanatory power. For this reason the GB account presented here seems to be preferable.

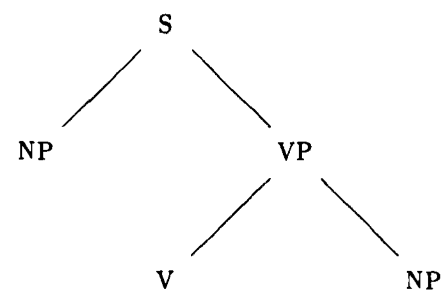


Figure 4.3: A TAG(LD/LP) for transitive sentences.

4.2 Shortcomings and Future Work

Perhaps the most obvious shortcoming of the currently implemented parser is the limited range of the Warlpiri language that it can parse. Warlpiri, like any other natural language, contains many intricate phenomena, and this parser has only begun to truly analyze it. In the first section below I list some outstanding constructions that deserve attention, and discuss how the parser might be modified to handle them.

The other arena in which the parser comes up short is the ability to parse more than one language. GB is, after all, a theory that attempts to explain Universal Grammar, and so a parser based on it should be able to parse many languages, not just one. The second section discusses what seems to be involved in extending the parser to cover other languages.

4.2.1 More Warlpiri

There is much more to Warlpiri than the picture presented in this thesis.[2] Table 4.2 gives a list of some of the remaining phenomena for the parser. Those listed in the first group are the most likely to be covered with relatively little effort. The second group shows a serious shortcoming of the parser that must be addressed before the parser can be said to properly parse natural language. The last group lists other phenomena that will demand a non-trivial amount of work, both linguistic and computational. However, even for these phenomena, the required modifications to the parser should follow in the same vein as the currently implemented system; no major rework of the basic engine seems necessary.

Perhaps the most tantalizing phenomenon is parsing continuous case phrases with intermediate case-marking. This is demonstrated in (5), repeated from the introductory chapter. The case phrase consists of two simpler case phrases, *marlu-ku* and *pukurlpa wiri-ki*, both of which exist in a single phonological phrase. Handling this phenomenon should involve no more than giving case-markers the phrasal action

[2]See [Nas86] for an extensive discussion of the phenomena of Warlpiri.

- continuous case phrase with intermediate case-marking
- discontinuous case phrases
- headship in multi-noun case phrases
- auxiliary complementizers

- lexical ambiguity
- structural ambiguity
- morphological ambiguity

- preverbs
- compounding
- topicalization
- infinitival clauses
- secondary predication
- nominal sentences

Table 4.2: Some unhandled Warlpiri phenomena.

of selecting an unbounded number of other case phrases. With this action, such case phrases would be parsed by first parsing each of the simpler continuous case phrases (*i.e.*, those with some number of unmarked nouns followed by a case-marked noun), and then grouping these case phrases with the newly added action.

(5) *marlu-ku pukurlpa wiri-ki*
kangaroo-DAT friendly big-DAT
'to/from the big, friendly kangaroo'

Parsing discontinuous case phrases also seems to be close at hand. (6) shows a variation of the sample sentence from the introduction with such a phrase, composed of the words *kurdu-ku* and *wiri-ki*. They are interpreted in unison, roughly as 'the big child', despite the separation in the sentence. The action to be added here would show up in syntax, so as to ignore the effects of ordering. Roughly speaking, case markers would be allowed to take similarly marked case phrases as arguments. But the precise structures to be derived are not so clear when the analyzing the head of the phrase is considered. That is, one noun of a case phrase is interpreted as the head of the phrase, with the other nouns acting as modifiers (much like adjectives in English). The determination of the head of a phrase and the modificational structure does not seem to be so straightforward, yet the ingredients for parsing full case phrases seem to be available. All that is needed is the theoretical recipe.

(6) *Ngajulu-rlu ka-rna-rla punta-rni kurdu-ku karli wiri-ki.*
I-ERG IMPERF-1s-3d take-NONPAST child-DAT
boomerang big-DAT
'I am taking the boomerang from the big child.'

As mentioned above, there is another element to the auxiliary word, namely, complementizers. These elements indicate the mood of the sentence and combine with the tense and aspect information. Morphologically speaking, they appear as the first element in the linear template of the auxiliary, so this form of parsing won't require too much effort. Syntactically speaking they don't present much of a problem either, as they can be entered into the auxiliary projection.

The next area on the list is the traditional problem of ambiguity which the current version of the parser assumes away. There are two flavors of ambiguity that fall under the purview of the parser. Lexical ambiguity exists when a single morpheme maps to more than one lexical entry, due to differing uses in syntax. Syntactic ambiguity occurs when more than one structure may be derived in the parse of a single sentence. At present I can offer no better solution other than the standard methods, such as simulation of a non-deterministic parser, or installation of a lookahead device to remove local ambiguity. This area of parsing deserves more attention.

Morphological ambiguity, the third type listed, will arise when the parser is extended to handle unseparated words. That is, instead of supplying sentences with sentence, phrase, word, and morpheme boundaries, only the former three will need to be supplied. The job of the parser then expands to breaking up each word into its constituent morphemes. For example, the word *ngajulurlu* can be broken into two *morpheme covers*, shown in (7). (As it happens, the first cover is ungrammatical, and only the second—the one used in the sample sentence—is grammatical.) Ambiguity exists when more than one cover is possible, such as the case given here. Again, one solution that comes to mind is to try each cover separately (*i.e.*, by simulating a non-deterministic parser), and halting the parse on those covers that do not pan out for some reason (*e.g.*, failing to combine morphologically).

(7) a. (*ngaju*, *lu*, *rlu*)
b. (*ngajulu*, *rlu*)

The last group in the list indicates the wide range of constructions in Warlpiri. Preverbs cliticize onto verb stems and affect the meaning and subcategorization of the predicator. For example, the preverb, *ngayi*, adds a benefactive aspect to the verb. Note that the change in meaning licenses another argument, which would also appear with dative case-marking. (8) shows the sample sentence from the introduction with the preverb added to the main verb, and the extra argument for which it subcategorizes. If preverbs were this simple, it wouldn't be too difficult to make the extensions for handling them. However, preverbs enter into many other constructions that are ill-understood.

(8) *Ngajulu-rlu ka-rna-rla-jinta ngayi-punta-rni kurdu-ku karli wati-ki.*
I-ERG IMPERF-1s-3d-DAT benefactive-take-NONPAST child-DAT
boomerang man-DAT
'I am taking the boomerang from the child for (the benefit of) the men.'

Compounding is a morphological phenomenon where two words combine to form a single word, with a corresponding change in meaning. For example, *marna* 'grass'

and *ngarnu* 'eat' can be compounded to form *marna-ngarnu* meaning a grass-eater. The morphology of these words is similar to that of case-marking, yet there are differences which will entail modifications to the parser.

Topicalization is the process of uttering a phrase before the sentence, often with a significant pause in between the fronted phrase and the sentence proper, for the purpose of emphasis. This is a stronger form than moving a phrase to the first position. In fact, there can be several topicalized phrases. Furthermore, topicalization will usually involve repetition of phrases rather than movement, and so the copied phrases must be identified during parsing. This is exemplified with another variation of the sample sentence, shown in (9). The syntax of topics does not seem too hard to state, but the real problem lies in its semantics. The theory of focus and emphasis is still quite impoverished.

(9) *Ngajulu-rlu, ngajulu-rlu ka-rna-rla punta-rni kurdu-ku karli.*
I-ERG I-ERG IMPERF-1s-3d take-NONPAST child-DAT boomerang
'As for me, I am taking the boomerang from the child.'

Infinitival clauses, like in English, are subordinate clauses that contain a verbal element and arguments. Often one of these argument is linked to one the arguments in the main phrase. (10) shows an example of an infinitival clause (in this case, controlled by the subject of the main clause), taken from the discussion of control theory in the second chapter. Note there is no multi-level embedding of subordinate clauses in Warlpiri; infinitival clauses may be nested just once in a sentence. The syntax of these clauses seems very similar to that of main clauses, and so it shouldn't be too difficult to extend the parser. Perhaps the trickiest part of this phenomenon are the issues of control which will require modification.

(10) *Karnta ka-ju wangka-mi yarla karla-nja-karra.*
woman IMPERF-1o speak-NONPAST yam dig-INF-COMP
'The woman is speaking to me while digging yams.'

Secondary predicates are like restrictive relative clauses in English which modify one of the main arguments of a sentence. As one might expect, these clauses are identified with the argument they modify by case-marking (rather than by position, as with English). In (11), the secondary clause modifies the subject, indicating the body part by which the action was achieved. The syntax of secondary predicates is like that of infinitival clauses in that there is an argument outside the clause that is the subject of the clause itself. As a result, secondary predication should follow fairly quickly from the extensions needed for infinitival clauses.

(11) *Ngajulu-rlu ka-rna-rla punta-rni rdaka-rlu kurdu-ku karli.*
I-ERG IMPERF-1s-3d take-NONPAST hand-ERG child-DAT
boomerang
'I am taking the boomerang from the child with my hands.'

The last phenomenon on the list, nominal sentences, is rather common in languages of the world. Nominal sentences are often used to state a feature about

something. They cannot be used to discuss actions, for instance, as would be done with verbal sentences. (12) gives an example of a nominal sentence. Despite the lack of a verb, this sentence can be interpreted, and it is given the reading that a copular provides in languages like English. As with some of the other phenomena listed above, the simpler cases of nominal sentences don't seem too far off for the parser. But once the real complexity of these sentences is encountered, their entry into the parser's repertoire becomes a bit more distant.

(12) *Ngarrka-jarra-pala wiri-jarra.*
man-DUAL-33s big-DUAL
'The two men are big.'

4.2.2 Other Languages

With only one language in the parser's domain, it is rather easy to choose among the remaining languages of the world for others to be parsed. This thesis has hinted at how English might be parsed, and that, in fact, is the next project I intend to undertake with the parser. Happily, it seems that only a small amount of modification will be needed.

The key to tailoring the parser to English is to recognize the correspondences between the Warlpiri phenomena and their English counterparts. First, consider morphology. Verbal morphology in English seems to be as simple as Warlpiri, with the tense element affixing onto the verb stem. Of course, there are many more exceptions with English verbs, but these can be dealt with later.

English nominal morphology is simpler because there are no case-markers (disregarding the genitive case, for instance). Thus, only nouns and number-markers (*i.e.*, "-s") will have to be covered.

Dative case-marking in English is performed at the word level and from left-to-right, as opposed to the morphological level and from right-to-left. This distinction will be simple to encode in the lexicon. (The prepositions "to" or "from" are the main dative case-markers in English.)

Nominative case-marking is performed by INFL (the English equivalent of AUX), and absolutive case-marking is performed by the verb. The verb follows suit in its case-marking direction of left-to-right, but INFL seems to mark its case in the other way. This seeming contradiction is already handled by the parser, as it processes the auxiliary in a special manner anyhow. In English, the auxiliary will simply mark its case right-to-left, rather then left-to-right as in Warlpiri.

As for the syntactic structures, they will be very similar to those used for Warlpiri, as mentioned in the introductory chapter. One discrepancy concerns the position of the subject. In the Warlpiri GB literature, the subject is placed in the specifier position of the verbal projection, whereas in the mainstream GB theory (which has most often focused on English), the subject is placed in the specifier position of INFL. The parser is already powerful enough to encode the distinction, but the theoretical differences should be ironed out so that a more unified structure can be used.

This is only a very rough sketch of how to build a corresponding parser for English, but it should serve to indicate the relative simplicity of the task. Of course, it remains to be done, but the parser looks like it will prove robust enough for the job.

Appendix A

Test Cases

A.1 Implementation Notes

Just a couple of notes about the implementation. First I should mention that the parser is actually quite fast. It takes about one to two seconds to parse the sample sentence, and not much longer to interpret the resulting syntactic structure to obtain the θ-role/word mapping.

The code itself is about 50 pages in length. There are 19 objects that comprise the program. The major objects are the precedence parser and the syntactic parser, as well as the lexicon. The precedence parser consists of four objects for each of the phonological levels, and one central parser containing the basic engine. The syntactic parser is a single object. Both parsers are based on the phrase-marker object that implements a simple forest structure; the precedence parser imposes ordering on the forest, whereas the syntactic parser does not. The phrase-marker, in turn, refers to category objects that implement the nodes of the parse tree. Categories contain the data and actions particular to the parser type. The lexicon, on the other hand, is constructed as a mapping of morphemes to lexical entries, themselves objects in the system. At the base of the system are five support objects implementing lists, sets, mappings, functions and arrays.

The program was written in the McFlavor system on a Symbolics 3600-series Lisp Machine (under release 6.1). The McFlavor system is an object-oriented flavor system written at MIT by Edward Barton, which is very much like the Lisp Machine flavor system. McFlavor was chosen because it runs in Maclisp (under TOPS20), as well as the Symbolics.

A.2 Tests Cases

This section contains a (rather long) series of test cases for the parser. The test types are listed in table A.1. For each type, a number of tests were conducted, and they are listed in their corresponding section. Ungrammatical inputs are labelled with an asterisk; note that all of them have, indeed, been declared ill-formed by the parser. Grammatical inputs are presented without annotation, and note that the parser has properly processed them too.

- verb stems
- verbs
- noun composition
- verb composition
- auxiliary composition
- continuous case phrases
- auxiliary positioning
- free phrase order
- argument identification
- null auxiliary components
- null anaphora
- too many arguments
- case marking
- auxiliary base agreement
- nominal agreement

Table A.1: Test cases.

A.2.1 Verb Stems

```
Parsing YA.

PS: 0: data: CONJUGATION: 5
       morpheme: YA
       category: VERB

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: THETA-ASSIGN: THEME
              category: VERB
              children: projection?: T
                        actions: SPECIFIER: (CASE-ASSIGN .
                                             ABSOLUTIVE)
                        data: THETA-ROLES: (THEME)
                              THEME: ABSOLUTIVE
                              SUBJECT: THEME
                        morpheme: YA
                        category: VERB


Parsing YULKA.

PS: 0: data: CONJUGATION: 1
       morpheme: YULKA
```

```
      category: VERB

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: THETA-ASSIGN: THEME
              category: VERB
              children: projection?: T
                        actions: THETA-LINK: PATH
                                 SPECIFIER: (CASE-ASSIGN .
                                             ABSOLUTIVE)
                        data: THETA-ROLES: (THEME PATH)
                              THEME: ABSOLUTIVE
                              PATH: DATIVE
                              SUBJECT: THEME
                              OBJECT: PATH
                        morpheme: YULKA
                        category: VERB

Parsing WARRI.

PS: 0: data: CONJUGATION: 2
       morpheme: WARRI
       category: VERB

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: THETA-LINK: PATH
                        data: THETA-ROLES: (AGENT PATH)
                              AGENT: ERGATIVE
                              PATH: DATIVE
                              SUBJECT: AGENT
                              OBJECT: PATH
                        morpheme: WARRI
                        category: VERB

Parsing NYA.

PS: 0: data: CONJUGATION: 3
       morpheme: NYA
```

```
      category: VERB

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: THETA-ASSIGN: THEME
                                 COMPLEMENT: (CASE-ASSIGN .
                                              ABSOLUTIVE)
                        data: THETA-ROLES: (AGENT THEME)
                              AGENT: ERGATIVE
                              THEME: ABSOLUTIVE
                              SUBJECT: AGENT
                              OBJECT: THEME
                        morpheme: NYA
                        category: VERB

Parsing PUNTA.

PS: 0: data: CONJUGATION: 2
       morpheme: PUNTA
       category: VERB

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: THETA-LINK: PATH
                                 THETA-ASSIGN: THEME
                                 COMPLEMENT: (CASE-ASSIGN .
                                              ABSOLUTIVE)
                        data: THETA-ROLES: (AGENT THEME PATH)
                              AGENT: ERGATIVE
                              THEME: ABSOLUTIVE
                              PATH: DATIVE
                              SUBJECT: AGENT
                              OBJECT: PATH
                        morpheme: PUNTA
                        category: VERB
```

A.2.2 Inflected Verbs

```
Parsing (YA NI).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 5
                    morpheme: YA
                    category: VERB

                 1: data: CONJUGATION: 5
                    morpheme: NI
                    category: TENSE

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ABSOLUTIVE
                       THETA-ASSIGN: THEME
              category: VERB
              children: projection?: T
                        data: TENSE: NONPAST
                              THETA-ROLES: (THEME)
                              THEME: ABSOLUTIVE
                              SUBJECT: THEME
                        morpheme: YA
                        category: VERB

Parsing (YULKA MI).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 1
                    morpheme: YULKA
                    category: VERB

                 1: data: CONJUGATION: 1
                    morpheme: MI
                    category: TENSE

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ABSOLUTIVE
                       THETA-ASSIGN: THEME
              category: VERB
              children: projection?: T
                        actions: THETA-LINK: PATH
```

```
                     data: TENSE: NONPAST
                           THETA-ROLES: (THEME PATH)
                           THEME: ABSOLUTIVE
                           PATH: DATIVE
                           SUBJECT: THEME
                           OBJECT: PATH
                     morpheme: YULKA
                     category: VERB

Parsing (WARRI RNI).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: WARRI
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNI
                    category: TENSE

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: LICENSE: (CASE-ASSIGN . ABSOLUTIVE)
                                 THETA-LINK: PATH
                        data: TENSE: NONPAST
                              THETA-ROLES: (AGENT PATH)
                              AGENT: ERGATIVE
                              PATH: DATIVE
                              SUBJECT: AGENT
                              OBJECT: PATH
                        morpheme: WARRI
                        category: VERB

Parsing (NYA NYI).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 3
                    morpheme: NYA
                    category: VERB
```

```
                    1: data: CONJUGATION: 3
                       morpheme: NYI
                       category: TENSE

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: CASE-ASSIGN: ABSOLUTIVE
                                 THETA-ASSIGN: THEME
                        data: TENSE: NONPAST
                              THETA-ROLES: (AGENT THEME)
                              AGENT: ERGATIVE
                              THEME: ABSOLUTIVE
                              SUBJECT: AGENT
                              OBJECT: THEME
                        morpheme: NYA
                        category: VERB

Parsing (PUNTA RNI).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNI
                    category: TENSE

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: CASE-ASSIGN: ABSOLUTIVE
                                 THETA-LINK: PATH
                                 THETA-ASSIGN: THEME
                        data: TENSE: NONPAST
                              THETA-ROLES: (AGENT THEME PATH)
```

```
                    AGENT: ERGATIVE
                    THEME: ABSOLUTIVE
                    PATH: DATIVE
                    SUBJECT: AGENT
                    OBJECT: PATH
              morpheme: PUNTA
              category: VERB
```

A.2.3 Noun Composition

```
Parsing (NGAJULU RLU).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

SS: projection?: NIL
    category: CASE
    children: projection?: NIL
              data: PERSON: 1
                    NUMBER: (SINGULAR)
              morpheme: NGAJULU
              category: NOUN

              projection?: T
              data: CASE-MARKED: ERGATIVE
              morpheme: RLU
              category: CASE

Parsing *(KARLI RNI).

(KARLI RNI) is ungrammatical.
The precedence structure is unconnected.

Parsing *(KU KURDU).

(KU KURDU) is ungrammatical.
The precedence structure is unconnected.

Parsing *(MARLU KU RLU).

(MARLU KU RLU) is ungrammatical.
```

The precedence structure is unconnected.

A.2.4 Verb Composition

```
Parsing (PUNTA RNI).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNI
                    category: TENSE

SS: projection?: NIL
    category: VERB
    children: projection?: T
              actions: CASE-ASSIGN: ERGATIVE
                       THETA-ASSIGN: AGENT
              category: VERB
              children: projection?: T
                        actions: CASE-ASSIGN: ABSOLUTIVE
                                 THETA-LINK: PATH
                                 THETA-ASSIGN: THEME
                        data: TENSE: NONPAST
                              THETA-ROLES: (AGENT THEME PATH)
                              AGENT: ERGATIVE
                              THEME: ABSOLUTIVE
                              PATH: DATIVE
                              SUBJECT: AGENT
                              OBJECT: PATH
                        morpheme: PUNTA
                        category: VERB

Parsing *(NYA KI).

(NYA KI) is ungrammatical.
The precedence structure is unconnected.

Parsing *(KU YULKA).

(KU YULKA) is ungrammatical.
The precedence structure is unconnected.
```

A.2.5 Auxiliary Composition

```
Parsing *(RNA).

(RNA) is ungrammatical.
The auxiliary has too few syllables.

Parsing *(RLA).

(RLA) is ungrammatical.
The auxiliary has too few syllables.

Parsing *(RNA RLA).

(RNA RLA) is ungrammatical.
The word begins with a clitic.

Parsing *(KA).

(KA) is ungrammatical.
The auxiliary has too few syllables.

Parsing (KA RNA).

PS: 0: category: AUXILIARY-BASE
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                                     AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-OBJECT
                                        AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: T
              actions: ARGUMENT: VERB
              category: AUXILIARY-BASE
              children: projection?: NIL
```

```
                    data: PERSON: 1
                          NUMBER: SINGULAR
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                    projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE
```

Parsing (KA RLA).

```
PS: 0: category: AUXILIARY-BASE
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                                     AUXILIARY-SELECT:
                                       AUXILIARY-SUBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: T
              actions: ARGUMENT: VERB
              category: AUXILIARY-BASE
              children: projection?: NIL
                        morpheme: RLA
                        category: AUXILIARY-DATIVE

                        projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE
```

Parsing (KA RNA RLA).

```
PS: 0: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
```

```
                                   AUXILIARY-OBJECT
                 data: SYLLABLES: 1
                 morpheme: KA
                 category: AUXILIARY-BASE

              1: data: SYLLABLES: 1
                 morpheme: RNA
                 category: AUXILIARY-SUBJECT

              2: data: SYLLABLES: 1
                 morpheme: RLA
                 category: AUXILIARY-DATIVE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: T
              actions: ARGUMENT: VERB
              category: AUXILIARY-BASE
              children: projection?: NIL
                        morpheme: RLA
                        category: AUXILIARY-DATIVE

                        projection?: NIL
                        data: PERSON: 1
                              NUMBER: SINGULAR
                        morpheme: RNA
                        category: AUXILIARY-SUBJECT

                        projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE

Parsing *(LPA RNA RLA).

(LPA RNA RLA) is ungrammatical.
The word begins with a clitic.

Parsing *(RNA KA).

(RNA KA) is ungrammatical.
The precedence structure is unconnected.

Parsing *(KA LPA).
```

```
(KA LPA) is ungrammatical.
The precedence structure is unconnected.

Parsing *(KA RNA RNA).

(KA RNA RNA) is ungrammatical.
The precedence structure is unconnected.
```

A.2.6 Continuous Case Phrases

```
Parsing ((YIRRINJI) (YIRRARU) (KARDIRRPA RLU)).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: YIRRINJI
                    category: NOUN

                 1: category: CASE
                    children: 0: morpheme: YIRRARU
                                 category: NOUN

                              1: category: CASE
                                 children: 0: morpheme: KARDIRRPA
                                              category: NOUN

                                           1: morpheme: RLU
                                              category: CASE

SS: projection?: NIL
    category: CASE
    children: projection?: NIL
              morpheme: YIRRINJI
              category: NOUN

              projection?: NIL
              morpheme: YIRRARU
              category: NOUN

              projection?: NIL
              morpheme: KARDIRRPA
              category: NOUN

              projection?: T
              data: CASE-MARKED: ERGATIVE
              morpheme: RLU
```

```
            category: CASE

Parsing ((YIRRINJI) (YIRRARU) (KARDIRRPA)).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: YIRRINJI
                    category: NOUN

                 1: category: CASE
                    children: 0: morpheme: YIRRARU
                                 category: NOUN

                              1: category: CASE
                                 children: 0: morpheme: KARDIRRPA
                                              category: NOUN

                                           1: morpheme: *ABS*
                                              category: CASE

SS: projection?: NIL
    category: CASE
    children: projection?: NIL
              morpheme: YIRRINJI
              category: NOUN

              projection?: NIL
              morpheme: YIRRARU
              category: NOUN

              projection?: NIL
              morpheme: KARDIRRPA
              category: NOUN

              projection?: T
              data: CASE-MARKED: ABSOLUTIVE
              morpheme: *ABS*
              category: CASE

Parsing *((YIRRINJI) (YIRRARU RLU) (KARDIRRPA)).

((YIRRINJI) (YIRRARU RLU) (KARDIRRPA)) is ungrammatical.
The precedence structure is unconnected.

Parsing *((YIRRINJI) (NYA NYI) (KARDIRRPA)).
```

```
((YIRRINJI) (NYA NYI) (KARDIRRPA)) is ungrammatical.
The precedence structure is unconnected.
```

A.2.7 Auxiliary Positioning

```
Parsing (((MARLU KA)) ((YA NI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: MARLU
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE

    1: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                        AUXILIARY-SELECT: AUXILIARY-OBJECT
                        AUXILIARY-SELECT: AUXILIARY-SUBJECT
                        RIGHT-ADJACENT: (AUXILIARY-SUBJECT
                                         AUXILIARY-OBJECT
                                         AUXILIARY-DATIVE)
       data: SYLLABLES: 1
       morpheme: KA
       category: AUXILIARY-BASE

    2: category: VERB
       children: 0: data: CONJUGATION: 5
                    morpheme: YA
                    category: VERB

                 1: data: CONJUGATION: 5
                    morpheme: NI
                    category: TENSE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: THEME
                              CASE-ASSIGNED: ABSOLUTIVE
                        category: CASE
                        children: projection?: NIL
                                  morpheme: MARLU
```

```
                              category: NOUN

                              projection?: T
                              data: CASE-MARKED: ABSOLUTIVE
                              morpheme: *ABS*
                              category: CASE

                  projection?: T
                  category: VERB
                  children: projection?: T
                            data: TENSE: NONPAST
                                  THETA-ROLES: (THEME)
                                  THEME: ABSOLUTIVE
                                  SUBJECT: THEME
                            morpheme: YA
                            category: VERB

      projection?: T
      category: AUXILIARY-BASE
      children: projection?: T
                data: TENSES: (NONPAST)
                      ASPECT: IMPERFECT
                morpheme: KA
                category: AUXILIARY-BASE

Parsing *(((MARLU) (KA)) ((YA NI))).

(KA) is ungrammatical.
The auxiliary has too few syllables.

Parsing *(((MARLU)) ((KA)) ((YA NI))).

(KA) is ungrammatical.
The auxiliary has too few syllables.

Parsing (((MARLU) (KA LU)) ((YA NI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: MARLU
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE
```

```
    1: category: AUXILIARY-BASE
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                                     AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-OBJECT
                                        AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: LU
                    category: AUXILIARY-SUBJECT

    2: category: VERB
       children: 0: data: CONJUGATION: 5
                    morpheme: YA
                    category: VERB

                 1: data: CONJUGATION: 5
                    morpheme: NI
                    category: TENSE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: THEME
                              CASE-ASSIGNED: ABSOLUTIVE
                        category: CASE
                        children: projection?: NIL
                                  morpheme: MARLU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ABSOLUTIVE
                                  morpheme: *ABS*
                                  category: CASE

                        projection?: T
                        category: VERB
```

```
                    children: projection?: T
                              data: TENSE: NONPAST
                                    THETA-ROLES: (THEME)
                                    THEME: ABSOLUTIVE
                                    SUBJECT: THEME
                              morpheme: YA
                              category: VERB

          projection?: T
          category: AUXILIARY-BASE
          children: projection?: NIL
                    data: PERSON: 3
                          NUMBER: PLURAL
                    morpheme: LU
                    category: AUXILIARY-SUBJECT

                    projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE

Parsing (((MARLU)) ((KA LU)) ((YA NI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: MARLU
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE

    1: category: AUXILIARY-BASE
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                                     AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-OBJECT
                                        AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
```

```
                morpheme: LU
                category: AUXILIARY-SUBJECT

     2: category: VERB
        children: 0: data: CONJUGATION: 5
                     morpheme: YA
                     category: VERB

                  1: data: CONJUGATION: 5
                     morpheme: NI
                     category: TENSE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: THEME
                              CASE-ASSIGNED: ABSOLUTIVE
                        category: CASE
                        children: projection?: NIL
                                  morpheme: MARLU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ABSOLUTIVE
                                  morpheme: *ABS*
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: T
                                  data: TENSE: NONPAST
                                        THETA-ROLES: (THEME)
                                        THEME: ABSOLUTIVE
                                        SUBJECT: THEME
                                  morpheme: YA
                                  category: VERB

              projection?: T
              category: AUXILIARY-BASE
              children: projection?: NIL
                        data: PERSON: 3
                              NUMBER: PLURAL
```

```
                    morpheme: LU
                    category: AUXILIARY-SUBJECT

                    projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE

Parsing (((KA LU)) ((MARLU)) ((YA NI))).

PS: 0: category: AUXILIARY-BASE
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                                     AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-OBJECT
                                        AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: LU
                    category: AUXILIARY-SUBJECT

    1: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: MARLU
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE

    2: category: VERB
       children: 0: data: CONJUGATION: 5
                    morpheme: YA
                    category: VERB

                 1: data: CONJUGATION: 5
                    morpheme: NI
                    category: TENSE

SS: projection?: NIL
```

```
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: THEME
                              CASE-ASSIGNED: ABSOLUTIVE
                        category: CASE
                        children: projection?: NIL
                                  morpheme: MARLU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ABSOLUTIVE
                                  morpheme: *ABS*
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: T
                                  data: TENSE: NONPAST
                                        THETA-ROLES: (THEME)
                                        THEME: ABSOLUTIVE
                                        SUBJECT: THEME
                                  morpheme: YA
                                  category: VERB

              projection?: T
              category: AUXILIARY-BASE
              children: projection?: NIL
                        data: PERSON: 3
                              NUMBER: PLURAL
                        morpheme: LU
                        category: AUXILIARY-SUBJECT

                        projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE

Parsing *(((MARLU)) ((YA NI)) ((KA LU))).

(((MARLU)) ((YA NI)) ((KA LU))) is ungrammatical.
The auxiliary is not in the proper position.
```

A.2.8 Free Phrase Order

```
Parsing (((NGAJULU RLU) (KA RNA RLA)) ((PUNTA RNI)) ((KURDU KU))
        ((KARLI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: data: SYLLABLES: 1
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

    2: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNI
                    category: TENSE

    3: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN

                 1: morpheme: KU
                    category: CASE

    4: phrasal actions: SELECT*: NOUN
```

```
        category: CASE
        children: 0: morpheme: KARLI
                     category: NOUN

                  1: morpheme: *ABS*
                     category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KARLI
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE

                                  projection?: NIL
                                  data: THETA-LINKED: PATH
```

```
                    category: CASE
                    children: projection?: NIL
                              morpheme: KURDU
                              category: NOUN

                              projection?: T
                              data: CASE-ASSIGNED:
                                      DATIVE
                                    CASE-MARKED:
                                      DATIVE
                                    THETA-ASSIGNED:
                                      PATH
                              morpheme: KU
                              category: CASE

                    projection?: T
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME
                                        PATH)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          PATH: DATIVE
                          SUBJECT: AGENT
                          OBJECT: PATH
                    morpheme: PUNTA
                    category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
          morpheme: RLA
          category: AUXILIARY-DATIVE

          projection?: NIL
          data: PERSON: 1
                NUMBER: SINGULAR
          morpheme: RNA
          category: AUXILIARY-SUBJECT

          projection?: T
          data: TENSES: (NONPAST)
                ASPECT: IMPERFECT
          morpheme: KA
          category: AUXILIARY-BASE
```

```
Parsing (((NGAJULU RLU) (KA RNA RLA)) ((KURDU KU)) ((KARLI))
        ((PUNTA RNI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: data: SYLLABLES: 1
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

    2: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN

                 1: morpheme: KU
                    category: CASE

    3: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KARLI
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE

    4: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
```

```
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNI
                    category: TENSE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KARLI
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE

                                  projection?: NIL
                                  data: THETA-LINKED: PATH
                                  category: CASE
```

```
                    children: projection?: NIL
                              morpheme: KURDU
                              category: NOUN

                              projection?: T
                              data: CASE-ASSIGNED:
                                      DATIVE
                                    CASE-MARKED:
                                      DATIVE
                                    THETA-ASSIGNED:
                                      PATH
                              morpheme: KU
                              category: CASE

                    projection?: T
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME
                                          PATH)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          PATH: DATIVE
                          SUBJECT: AGENT
                          OBJECT: PATH
                    morpheme: PUNTA
                    category: VERB

          projection?: T
          category: AUXILIARY-BASE
          children: projection?: NIL
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

                    projection?: NIL
                    data: PERSON: 1
                          NUMBER: SINGULAR
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                    projection?: T
                    data: TENSES: (NONPAST)
                          ASPECT: IMPERFECT
                    morpheme: KA
                    category: AUXILIARY-BASE

Parsing (((KURDU KU) (KA RNA RLA)) ((PUNTA RNI)) ((NGAJULU RLU))
```

```
        ((KARLI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN

                 1: morpheme: KU
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: data: SYLLABLES: 1
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

    2: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNI
                    category: TENSE

    3: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    4: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KARLI
```

```
                 category: NOUN

              1: morpheme: *ABS*
                 category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KARLI
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE

                                  projection?: NIL
                                  data: THETA-LINKED: PATH
                                  category: CASE
                                  children: projection?: NIL
```

```
                                morpheme: KURDU
                                category: NOUN

                                projection?: T
                                data: CASE-ASSIGNED:
                                        DATIVE
                                      CASE-MARKED:
                                        DATIVE
                                      THETA-ASSIGNED:
                                        PATH
                                morpheme: KU
                                category: CASE

                      projection?: T
                      data: TENSE: NONPAST
                            THETA-ROLES: (AGENT THEME
                                          PATH)
                            AGENT: ERGATIVE
                            THEME: ABSOLUTIVE
                            PATH: DATIVE
                            SUBJECT: AGENT
                            OBJECT: PATH
                      morpheme: PUNTA
                      category: VERB

    projection?: T
    category: AUXILIARY-BASE
    children: projection?: NIL
              morpheme: RLA
              category: AUXILIARY-DATIVE

              projection?: NIL
              data: PERSON: 1
                    NUMBER: SINGULAR
              morpheme: RNA
              category: AUXILIARY-SUBJECT

              projection?: T
              data: TENSES: (NONPAST)
                    ASPECT: IMPERFECT
              morpheme: KA
              category: AUXILIARY-BASE

Parsing (((KARLI) (KA RNA RLA)) ((NGAJULU RLU)) ((KURDU KU))
        ((PUNTA RNI))).
```

```
PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KARLI
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: data: SYLLABLES: 1
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

    2: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    3: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN

                 1: morpheme: KU
                    category: CASE

    4: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
                    category: VERB

                 1: data: CONJUGATION: 2
```

```
                  morpheme: RNI
                  category: TENSE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KARLI
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE

                                  projection?: NIL
                                  data: THETA-LINKED: PATH
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KURDU
                                            category: NOUN
```

```
                    projection?: T
                    data: CASE-ASSIGNED:
                            DATIVE
                          CASE-MARKED:
                            DATIVE
                          THETA-ASSIGNED:
                            PATH
                    morpheme: KU
                    category: CASE

          projection?: T
          data: TENSE: NONPAST
                THETA-ROLES: (AGENT THEME
                              PATH)
                AGENT: ERGATIVE
                THEME: ABSOLUTIVE
                PATH: DATIVE
                SUBJECT: AGENT
                OBJECT: PATH
          morpheme: PUNTA
          category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
          morpheme: RLA
          category: AUXILIARY-DATIVE

          projection?: NIL
          data: PERSON: 1
                NUMBER: SINGULAR
          morpheme: RNA
          category: AUXILIARY-SUBJECT

          projection?: T
          data: TENSES: (NONPAST)
                ASPECT: IMPERFECT
          morpheme: KA
          category: AUXILIARY-BASE
```

A.2.9 Argument Identification

Parsing (((NGAJULU RLU) (KA RNA RLA)) ((PUNTA RNI)) ((KURDU KU))
((KARLI))).

```
PATH: projection?: NIL
      morpheme: KURDU
      category: NOUN

THEME: projection?: NIL
       morpheme: KARLI
       category: NOUN

AGENT: projection?: NIL
       data: PERSON: 1
             NUMBER: (SINGULAR)
       morpheme: NGAJULU
       category: NOUN

Parsing (((NGAJULU RLU) (KA RNA RLA)) ((KURDU KU)) ((KARLI))
         ((PUNTA RNI))).

PATH: projection?: NIL
      morpheme: KURDU
      category: NOUN

THEME: projection?: NIL
       morpheme: KARLI
       category: NOUN

AGENT: projection?: NIL
       data: PERSON: 1
             NUMBER: (SINGULAR)
       morpheme: NGAJULU
       category: NOUN

Parsing (((KURDU KU) (KA RNA RLA)) ((PUNTA RNI)) ((NGAJULU RLU))
         ((KARLI))).

PATH: projection?: NIL
      morpheme: KURDU
      category: NOUN

THEME: projection?: NIL
       morpheme: KARLI
       category: NOUN

AGENT: projection?: NIL
       data: PERSON: 1
             NUMBER: (SINGULAR)
```

```
       morpheme: NGAJULU
       category: NOUN

Parsing (((KARLI) (KA RNA RLA)) ((NGAJULU RLU)) ((KURDU KU))
         ((PUNTA RNI))).

PATH: projection?: NIL
      morpheme: KURDU
      category: NOUN

THEME: projection?: NIL
       morpheme: KARLI
       category: NOUN

AGENT: projection?: NIL
       data: PERSON: 1
             NUMBER: (SINGULAR)
       morpheme: NGAJULU
       category: NOUN
```

A.2.10 Null Auxiliary Components

```
Parsing (((MARLU RLU KA)) ((NYA NYI)) ((KURDU))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: MARLU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                        AUXILIARY-SELECT: AUXILIARY-OBJECT
                        AUXILIARY-SELECT: AUXILIARY-SUBJECT
                        RIGHT-ADJACENT: (AUXILIARY-SUBJECT
                                         AUXILIARY-OBJECT
                                         AUXILIARY-DATIVE)
       data: SYLLABLES: 1
       morpheme: KA
       category: AUXILIARY-BASE

    2: category: VERB
       children: 0: data: CONJUGATION: 3
                    morpheme: NYA
                    category: VERB
```

```
                  1: data: CONJUGATION: 3
                     morpheme: NYI
                     category: TENSE

         3: phrasal actions: SELECT*: NOUN
            category: CASE
            children: 0: morpheme: KURDU
                         category: NOUN

                      1: morpheme: *ABS*
                         category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  morpheme: MARLU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KURDU
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE
```

```
                                projection?: T
                                data: TENSE: NONPAST
                                      THETA-ROLES: (AGENT THEME)
                                      AGENT: ERGATIVE
                                      THEME: ABSOLUTIVE
                                      SUBJECT: AGENT
                                      OBJECT: THEME
                                morpheme: NYA
                                category: VERB

              projection?: T
              category: AUXILIARY-BASE
              children: projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE

Parsing (((NGAJULU RLU KA RNA)) ((NYA NYI)) ((KURDU))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: category: AUXILIARY-BASE
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                                     AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-OBJECT
                                        AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

    2: category: VERB
```

```
      children: 0: data: CONJUGATION: 3
                   morpheme: NYA
                   category: VERB

                1: data: CONJUGATION: 3
                   morpheme: NYI
                   category: TENSE

   3: phrasal actions: SELECT*: NOUN
      category: CASE
      children: 0: morpheme: KURDU
                   category: NOUN

                1: morpheme: *ABS*
                   category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KURDU
                                            category: NOUN
```

```
                                        projection?: T
                                        data: CASE-MARKED:
                                                ABSOLUTIVE
                                        morpheme: *ABS*
                                        category: CASE

                              projection?: T
                              data: TENSE: NONPAST
                                    THETA-ROLES: (AGENT THEME)
                                    AGENT: ERGATIVE
                                    THEME: ABSOLUTIVE
                                    SUBJECT: AGENT
                                    OBJECT: THEME
                              morpheme: NYA
                              category: VERB

            projection?: T
            category: AUXILIARY-BASE
            children: projection?: NIL
                      data: PERSON: 1
                            NUMBER: SINGULAR
                      morpheme: RNA
                      category: AUXILIARY-SUBJECT

                      projection?: T
                      data: TENSES: (NONPAST)
                            ASPECT: IMPERFECT
                      morpheme: KA
                      category: AUXILIARY-BASE

Parsing (((NGAJULU RLU KA RNA NGKU)) ((NYA NYI)) ((NYUNTULU))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                    data: SYLLABLES: 1
                    morpheme: KA
```

```
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: NGKU
                    category: AUXILIARY-OBJECT

   2: category: VERB
      children: 0: data: CONJUGATION: 3
                   morpheme: NYA
                   category: VERB

                1: data: CONJUGATION: 3
                   morpheme: NYI
                   category: TENSE

   3: phrasal actions: SELECT*: NOUN
      category: CASE
      children: 0: morpheme: NYUNTULU
                   category: NOUN

                1: morpheme: *ABS*
                   category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
```

```
                    data: CASE-MARKED: ERGATIVE
                    morpheme: RLU
                    category: CASE

          projection?: T
          category: VERB
          children: projection?: NIL
                    data: THETA-ASSIGNED: THEME
                          CASE-ASSIGNED: ABSOLUTIVE
                    category: CASE
                    children: projection?: NIL
                              data: PERSON: 2
                                    NUMBER: (SINGULAR)
                              morpheme: NYUNTULU
                              category: NOUN

                              projection?: T
                              data: CASE-MARKED:
                                      ABSOLUTIVE
                              morpheme: *ABS*
                              category: CASE

                    projection?: T
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          SUBJECT: AGENT
                          OBJECT: THEME
                    morpheme: NYA
                    category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
          data: PERSON: 2
                NUMBER: SINGULAR
          morpheme: NGKU
          category: AUXILIARY-OBJECT

          projection?: NIL
          data: PERSON: 1
                NUMBER: SINGULAR
          morpheme: RNA
          category: AUXILIARY-SUBJECT
```

```
              projection?: T
              data: TENSES: (NONPAST)
                    ASPECT: IMPERFECT
              morpheme: KA
              category: AUXILIARY-BASE
```

Parsing *(((NGAJULU RLU KA NGKU)) ((NYA NYI)) ((NYUNTULU))).

(((NGAJULU RLU KA NGKU)) ((NYA NYI)) ((NYUNTULU))) is ungrammatical.
The DEFAULT-AUXILIARY clitic does not agree in person with the SUBJECT.

Parsing *(((NGAJULU RLU KA RNA)) ((NYA NYI)) ((NYUNTULU))).

(((NGAJULU RLU KA RNA)) ((NYA NYI)) ((NYUNTULU))) is ungrammatical.
The DEFAULT-AUXILIARY clitic does not agree in person with the OBJECT.

A.2.11 Null Anaphora

Parsing (((NYA NYI KA)) ((KURDU))).

```
PS: 0: category: VERB
       children: 0: data: CONJUGATION: 3
                    morpheme: NYA
                    category: VERB

                 1: data: CONJUGATION: 3
                    morpheme: NYI
                    category: TENSE

    1: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                        AUXILIARY-SELECT: AUXILIARY-OBJECT
                        AUXILIARY-SELECT: AUXILIARY-SUBJECT
                        RIGHT-ADJACENT: (AUXILIARY-SUBJECT
                                         AUXILIARY-OBJECT
                                         AUXILIARY-DATIVE)
       data: SYLLABLES: 1
       morpheme: KA
       category: AUXILIARY-BASE

    2: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN
```

```
                 1: morpheme: *ABS*
                    category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: T
                        actions: CASE-ASSIGN: ERGATIVE
                                 THETA-ASSIGN: AGENT
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KURDU
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE

                                  projection?: T
                                  data: TENSE: NONPAST
                                        THETA-ROLES: (AGENT THEME)
                                        AGENT: ERGATIVE
                                        THEME: ABSOLUTIVE
                                        SUBJECT: AGENT
                                        OBJECT: THEME
                                  morpheme: NYA
                                  category: VERB

              projection?: T
              category: AUXILIARY-BASE
              children: projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE

Parsing (((NYA NYI KA))).
```

```
PS: 0: category: VERB
       children: 0: data: CONJUGATION: 3
                    morpheme: NYA
                    category: VERB

                 1: data: CONJUGATION: 3
                    morpheme: NYI   .
                    category: TENSE

    1: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                        AUXILIARY-SELECT: AUXILIARY-OBJECT
                        AUXILIARY-SELECT: AUXILIARY-SUBJECT
                        RIGHT-ADJACENT: (AUXILIARY-SUBJECT
                                         AUXILIARY-OBJECT
                                         AUXILIARY-DATIVE)
       data: SYLLABLES: 1
       morpheme: KA
       category: AUXILIARY-BASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: T
                        actions: CASE-ASSIGN: ERGATIVE
                                 THETA-ASSIGN: AGENT
                        category: VERB
                        children: projection?: T
                                  actions: CASE-ASSIGN: ABSOLUTIVE
                                           THETA-ASSIGN: THEME
                                  data: TENSE: NONPAST
                                        THETA-ROLES: (AGENT THEME)
                                        AGENT: ERGATIVE
                                        THEME: ABSOLUTIVE
                                        SUBJECT: AGENT
                                        OBJECT: THEME
                                  morpheme: NYA
                                  category: VERB

              projection?: T
              category: AUXILIARY-BASE
              children: projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
```

```
                         morpheme: KA
                         category: AUXILIARY-BASE

Parsing (((KA RNA NGKU)) ((NYA NYI)) ((NYUNTULU))).

PS: 0: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-DATIVE
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: lexical actions: RIGHT-ADJACENT:
                                       (AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: NGKU
                    category: AUXILIARY-OBJECT

    1: category: VERB
       children: 0: data: CONJUGATION: 3
                    morpheme: NYA
                    category: VERB

                 1: data: CONJUGATION: 3
                    morpheme: NYI
                    category: TENSE

    2: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NYUNTULU
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: T
```

```
actions: CASE-ASSIGN: ERGATIVE
         THETA-ASSIGN: AGENT
category: VERB
children: projection?: NIL
          data: THETA-ASSIGNED: THEME
                CASE-ASSIGNED: ABSOLUTIVE
          category: CASE
          children: projection?: NIL
                    data: PERSON: 2
                          NUMBER: (SINGULAR)
                    morpheme: NYUNTULU
                    category: NOUN

                    projection?: T
                    data: CASE-MARKED:
                            ABSOLUTIVE
                    morpheme: *ABS*
                    category: CASE

          projection?: T
          data: TENSE: NONPAST
                THETA-ROLES: (AGENT THEME)
                AGENT: ERGATIVE
                THEME: ABSOLUTIVE
                SUBJECT: AGENT
                OBJECT: THEME
          morpheme: NYA
          category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
          data: PERSON: 2
                NUMBER: SINGULAR
          morpheme: NGKU
          category: AUXILIARY-OBJECT

          projection?: NIL
          data: PERSON: 1
                NUMBER: SINGULAR
          morpheme: RNA
          category: AUXILIARY-SUBJECT

          projection?: T
          data: TENSES: (NONPAST)
```

```
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE

Parsing (((NGAJULU RLU KA RNA NGKU)) ((NYA NYI))).

PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                      AUXILIARY-DATIVE
                    data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: lexical actions: RIGHT-ADJACENT:
                                      (AUXILIARY-DATIVE)
                    data: SYLLABLES: 1
                    morpheme: NGKU
                    category: AUXILIARY-OBJECT

    2: category: VERB
       children: 0: data: CONJUGATION: 3
                    morpheme: NYA
                    category: VERB

                 1: data: CONJUGATION: 3
                    morpheme: NYI
                    category: TENSE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
```

```
children: projection?: NIL
          data: THETA-ASSIGNED: AGENT
                CASE-ASSIGNED: ERGATIVE
          category: CASE
          children: projection?: NIL
                    data: PERSON: 1
                          NUMBER: (SINGULAR)
                    morpheme: NGAJULU
                    category: NOUN

                    projection?: T
                    data: CASE-MARKED: ERGATIVE
                    morpheme: RLU
                    category: CASE

          projection?: T
          category: VERB
          children: projection?: T
                    actions: CASE-ASSIGN: ABSOLUTIVE
                             THETA-ASSIGN: THEME
                    data: TENSE: NONPAST
                          THETA-ROLES: (AGENT THEME)
                          AGENT: ERGATIVE
                          THEME: ABSOLUTIVE
                          SUBJECT: AGENT
                          OBJECT: THEME
                    morpheme: NYA
                    category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
          data: PERSON: 2
                NUMBER: SINGULAR
          morpheme: NGKU
          category: AUXILIARY-OBJECT

          projection?: NIL
          data: PERSON: 1
                NUMBER: SINGULAR
          morpheme: RNA
          category: AUXILIARY-SUBJECT

          projection?: T
          data: TENSES: (NONPAST)
```

```
            ASPECT: IMPERFECT
      morpheme: KA
      category: AUXILIARY-BASE
```

A.2.12 Too Many Arguments

```
Parsing *(((YA NI KA)) ((NGAJULU RLU)) ((KURDU KU)) ((KARLI))).

(((YA NI KA)) ((NGAJULU RLU)) ((KURDU KU)) ((KARLI))) is
ungrammatical.
The syntactic structure is unconnected.

Parsing *(((YA NI KA)) ((KURDU KU)) ((KARLI))).

(((YA NI KA)) ((KURDU KU)) ((KARLI))) is ungrammatical.
The syntactic structure is unconnected.

Parsing (((YA NI KA)) ((KURDU))).

PS: 0: category: VERB
       children: 0: data: CONJUGATION: 5
                    morpheme: YA
                    category: VERB

                 1: data: CONJUGATION: 5
                    morpheme: NI
                    category: TENSE

    1: lexical actions: AUXILIARY-SELECT: AUXILIARY-DATIVE
                        AUXILIARY-SELECT: AUXILIARY-OBJECT
                        AUXILIARY-SELECT: AUXILIARY-SUBJECT
                        RIGHT-ADJACENT: (AUXILIARY-SUBJECT
                                         AUXILIARY-OBJECT
                                         AUXILIARY-DATIVE)
       data: SYLLABLES: 1
       morpheme: KA
       category: AUXILIARY-BASE

    2: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN

                 1: morpheme: *ABS*
                    category: CASE
```

```
SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: THEME
                              CASE-ASSIGNED: ABSOLUTIVE
                        category: CASE
                        children: projection?: NIL
                                  morpheme: KURDU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ABSOLUTIVE
                                  morpheme: *ABS*
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: T
                                  data: TENSE: NONPAST
                                        THETA-ROLES: (THEME)
                                        THEME: ABSOLUTIVE
                                        SUBJECT: THEME
                                  morpheme: YA
                                  category: VERB

              projection?: T
              category: AUXILIARY-BASE
              children: projection?: T
                        data: TENSES: (NONPAST)
                              ASPECT: IMPERFECT
                        morpheme: KA
                        category: AUXILIARY-BASE
```

A.2.13 Case Marking

```
Parsing (((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU))
        ((NYUNTULU KU))).

PS: 0: category: AUXILIARY-OBJECT
       children: 0: data: SYLLABLES: 1
                    morpheme: KA
                    category: AUXILIARY-BASE
```

```
                1: data: SYLLABLES: 1
                   morpheme: RNA
                   category: AUXILIARY-SUBJECT

                2: data: SYLLABLES: 1
                   morpheme: NGKU
                   category: AUXILIARY-OBJECT

                3: data: SYLLABLES: 1
                   morpheme: RLA
                   category: AUXILIARY-DATIVE

    1: category: VERB
       children: 0: data: CONJUGATION: 1
                   morpheme: YULKA
                   category: VERB

                1: data: CONJUGATION: 1
                   morpheme: MI
                   category: TENSE

    2: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                   category: NOUN

                1: morpheme: *ABS*
                   category: CASE

    3: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NYUNTULU
                   category: NOUN

                1: morpheme: KU
                   category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: THEME
                              CASE-ASSIGNED: ABSOLUTIVE
```

```
category: CASE
children: projection?: NIL
          data: PERSON: 1
                NUMBER: (SINGULAR)
          morpheme: NGAJULU
          category: NOUN

          projection?: T
          data: CASE-MARKED: ABSOLUTIVE
          morpheme: *ABS*
          category: CASE

projection?: T
category: VERB
children: projection?: NIL
          data: THETA-LINKED: PATH
          category: CASE
          children: projection?: NIL
                    data: PERSON: 2
                          NUMBER: (SINGULAR)
                    morpheme: NYUNTULU
                    category: NOUN

                    projection?: T
                    data: CASE-ASSIGNED:
                            DATIVE
                          CASE-MARKED: DATIVE
                          THETA-ASSIGNED:
                            PATH
                    morpheme: KU
                    category: CASE

          projection?: T
          data: TENSE: NONPAST
                THETA-ROLES: (THEME PATH)
                THEME: ABSOLUTIVE
                PATH: DATIVE
                SUBJECT: THEME
                OBJECT: PATH
          morpheme: YULKA
          category: VERB

projection?: T
category: AUXILIARY-BASE
children: projection?: NIL
```

```
morpheme: RLA
category: AUXILIARY-DATIVE

projection?: NIL
data: PERSON: 2
      NUMBER: SINGULAR
morpheme: NGKU
category: AUXILIARY-OBJECT

projection?: NIL
data: PERSON: 1
      NUMBER: SINGULAR
morpheme: RNA
category: AUXILIARY-SUBJECT

projection?: T
data: TENSES: (NONPAST)
      ASPECT: IMPERFECT
morpheme: KA
category: AUXILIARY-BASE
```

```
Parsing *(((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU RLU))
         ((NYUNTULU))).

(((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU RLU)) ((NYUNTULU))) is
ungrammatical.
The syntactic structure is unconnected.

Parsing *(((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU KU))
         ((NYUNTULU RLU))).

(((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU KU)) ((NYUNTULU RLU))) is
ungrammatical.
The syntactic structure is unconnected.
```

A.2.14 Auxiliary Base Agreement

```
Parsing *(((NGAJULU RLU LPA RNA RLA)) ((PUNTA RNI)) ((KURDU KU))
         ((KARLI))).

(((NGAJULU RLU LPA RNA RLA)) ((PUNTA RNI)) ((KURDU KU)) ((KARLI))) is
ungrammatical.
The tenses of LPA and PUNTA do not match.

Parsing (((NGAJULU RLU LPA RNA RLA)) ((PUNTA RNU)) ((KURDU KU))
        ((KARLI))).
```

```
PS: 0: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: NGAJULU
                    category: NOUN

                 1: morpheme: RLU
                    category: CASE

    1: category: AUXILIARY-SUBJECT
       children: 0: lexical actions: AUXILIARY-SELECT:
                                       AUXILIARY-OBJECT
                    data: SYLLABLES: 1
                    morpheme: LPA
                    category: AUXILIARY-BASE

                 1: data: SYLLABLES: 1
                    morpheme: RNA
                    category: AUXILIARY-SUBJECT

                 2: data: SYLLABLES: 1
                    morpheme: RLA
                    category: AUXILIARY-DATIVE

    2: category: VERB
       children: 0: data: CONJUGATION: 2
                    morpheme: PUNTA
                    category: VERB

                 1: data: CONJUGATION: 2
                    morpheme: RNU
                    category: TENSE

    3: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KURDU
                    category: NOUN

                 1: morpheme: KU
                    category: CASE

    4: phrasal actions: SELECT*: NOUN
       category: CASE
       children: 0: morpheme: KARLI
                    category: NOUN
```

```
                1: morpheme: *ABS*
                   category: CASE

SS: projection?: NIL
    category: AUXILIARY-BASE
    children: projection?: NIL
              data: ARGUMENT: VERB
              category: VERB
              children: projection?: NIL
                        data: THETA-ASSIGNED: AGENT
                              CASE-ASSIGNED: ERGATIVE
                        category: CASE
                        children: projection?: NIL
                                  data: PERSON: 1
                                        NUMBER: (SINGULAR)
                                  morpheme: NGAJULU
                                  category: NOUN

                                  projection?: T
                                  data: CASE-MARKED: ERGATIVE
                                  morpheme: RLU
                                  category: CASE

                        projection?: T
                        category: VERB
                        children: projection?: NIL
                                  data: THETA-ASSIGNED: THEME
                                        CASE-ASSIGNED: ABSOLUTIVE
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KARLI
                                            category: NOUN

                                            projection?: T
                                            data: CASE-MARKED:
                                                    ABSOLUTIVE
                                            morpheme: *ABS*
                                            category: CASE

                                  projection?: NIL
                                  data: THETA-LINKED: PATH
                                  category: CASE
                                  children: projection?: NIL
                                            morpheme: KURDU
                                            category: NOUN
```

```
                                    projection?: T
                                    data: CASE-ASSIGNED:
                                            DATIVE
                                          CASE-MARKED:
                                            DATIVE
                                          THETA-ASSIGNED:
                                            PATH
                                    morpheme: KU
                                    category: CASE

                              projection?: T
                              data: TENSE: PAST
                                    THETA-ROLES: (AGENT THEME
                                                  PATH)
                                    AGENT: ERGATIVE
                                    THEME: ABSOLUTIVE
                                    PATH: DATIVE
                                    SUBJECT: AGENT
                                    OBJECT: PATH
                              morpheme: PUNTA
                              category: VERB

                projection?: T
                category: AUXILIARY-BASE
                children: projection?: NIL
                          morpheme: RLA
                          category: AUXILIARY-DATIVE

                          projection?: NIL
                          data: PERSON: 1
                                NUMBER: SINGULAR
                          morpheme: RNA
                          category: AUXILIARY-SUBJECT

                          projection?: T
                          data: TENSES: (PAST IRREALIS)
                                ASPECT: IMPERFECT
                          morpheme: LPA
                          category: AUXILIARY-BASE
```

A.2.15 Nominal Agreement

```
Parsing *(((KA RNA NGKU RLA)) ((YULKA MI)) ((MARLU))
        ((NYUNTULU KU))).
```

```
(((KA RNA NGKU RLA)) ((YULKA MI)) ((MARLU)) ((NYUNTULU KU))) is
ungrammatical.
The AUXILIARY-SUBJECT clitic does not agree in person with the
SUBJECT.

Parsing *(((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU))
         ((YIRRINJI KI))).

(((KA RNA NGKU RLA)) ((YULKA MI)) ((NGAJULU)) ((YIRRINJI KI))) is
ungrammatical.
The AUXILIARY-OBJECT clitic does not agree in person with the OBJECT.
```

Bibliography

[Abn87] S. Abney. Licensing and Parsing. In *NELS 16*. North Eastern Linguistic Society, University of Massachusetts, Amherst, MA, 1987.

[Bre82] J. Bresnan. *The Mental Representation of Grammatical Relations*. MIT Press, Cambridge, MA, 1982.

[BW84] R.C. Berwick and A.S. Weinberg. *The Grammatical Basis of Linguistic Performance: Language Use and Acquisition*. Volume 11 of *Current Studies in Linguistics*, MIT Press, Cambridge, MA, 1984.

[Cho65] N. Chomsky. *Aspects of the Theory of Syntax*. MIT Press, Cambridge, MA, 1965.

[Cho81] N. Chomsky. *Lectures on Government and Binding, the Pisa Lectures*. Volume 9 of *Studies in Generative Grammar*, Foris Publications, Dordrecht, 1981.

[Cho82] N. Chomsky. *Some Concepts and Consequences of the Theory of Government and Binding*. Volume 6 of *Linguistic Inquiry Monograph*, MIT Press, Cambridge, MA, 1982.

[Cho86] N. Chomsky. *Knowledge of Language: Its Nature, Origin, and Use*. *Convergence*, Praeger, Westport, CT, 1986.

[Cur61] H.B. Curry. Some Logical Aspects of Grammatical Structure. In R. Jakobson, editor, *Structure of Language and Its Mathematical Aspects, Proceedings of the Twelfth Symposium in Applied Mathematics*, pages 56–68. American Mathematical Society, Providence, RI, 1961.

[Dor87] B. Dorr. *UNITRAN: A Principle-Based Approach to Machine Translation*. Master's thesis, Massachusetts Institute of Technology, 1987. Forthcoming.

[Emo76] J. Emonds. *A Transformational Approach to English Syntax*. Academic Press, London, 1976.

[Hal83] K. Hale. Warlpiri and the Grammar of Non-configurational Languages. *Natural Language and Linguistic Theory*, 1:5–47, 1983.

[Joh85] M. Johnson. Parsing with Discontinuous Constituents. In *23rd Annual Proceedings of the Association for Computational Linguistics*, pages 127–32, Assocication for Computational Linguistics, 1985.

[Jos] A. Joshi. The Convergence of Mildly Context-sensitive Formalisms for Natural Language Processing. Forthcoming.

[Kla82] J. Klavans. Configuration in Non-configurational Languages. In *Proceedings of the West Coast Conference on Formal Linguistics*, Stanford University, Stanford, CA, 1982.

[Lau78] M. Laughren. Directional Terminology in Warlpiri, a Central Australian Language. *Working Papers in Language and Linguistics*, 8:1–16, 1978. Tasmanian College of Advanced Education, Launceston.

[Lev85] B. Levin, editor. *Lexical Semantics in Review*. Center for Cognitive Science, Massachusetts Institute of Technology, 1985.

[Mar82] D. Marr. *Vision*. W.H. Freeman and Company, San Francisco, 1982.

[Nas86] D. Nash. *Topics in Warlpiri Grammar. Outstanding Dissertations in Linguistics*, Garland Publishing Inc., New York, 1986.

[Sha85] R.M. Sharp. *A Model of Grammar Based of Principles of Government and Binding*. Master's thesis, The University of British Columbia, October 1985.

[Sim83] J.H. Simpson. *Aspects of Warlpiri Morphology and Syntax*. PhD thesis, Massachusetts Institute of Technology, April 1983.

[Sto81] T.A. Stowell. *Origins of Phrase Structure*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA, September 1981.

[vRW86] H. van Riemsdijk and E. Williams. *Introduction to the Theory of Grammar*. Volume 12 of *Current Studies in Linguistics*, MIT Press, Cambridge, MA, 1986.

[Wac92] J. Wackernagel. Über ein Gesetz der indogermanischen Wortsellung. *Indogermanische Forschungen*, 1:333–436, 1892.

[Weh84] E. Wehrli. *A Government-Binding Parser for French*. Technical Report 48, Université de Genève, 1984.

[Wil81] E. Williams. Argument Structure and Morphology. *The Linguistic Review*, 1:81–114, 1981.

[Wil84] E. Williams. Grammatical Relations. *Linguistic Inquiry*, 15(4):639–73, Fall 1984.

END
DATE
FILMED
MARCH
1988
DTIC